

Taiser Tadeu Teixeira Barros

**Formação em Pensamento Computacional
utilizando *Scratch* para Professores de
Matemática e Informática da Educação
Fundamental**

Brasil

Novembro, 2020

Taiser Tadeu Teixeira Barros

**Formação em Pensamento Computacional utilizando
Scratch para Professores de Matemática e Informática
da Educação Fundamental**

Tese apresentada ao Programa de Pós-Graduação em Informática na Educação do Centro Interdisciplinar de Novas Tecnologias na Educação da Universidade Federal do Rio Grande do Sul, como requisito para obtenção do título de Doutor em Informática na Educação. Linha de pesquisa: Ambientes Informatizados e Ensino a Distância.

Universidade Federal do Rio Grande do Sul – UFRGS

Programa de Pós-Graduação em Informática na Educação – PPGIE

Orientador: Prof. Dr. Eliseo Berni Reategui

Coorientador: Prof. Dr. Adriano Canabarro Teixeira

Brasil

Novembro, 2020

CIP - Catalogação na Publicação

BARROS, TAISER TADEU TEIXEIRA
Formação em Pensamento Computacional utilizando
Scratch para Professores de Matemática e Informática
da Educação Fundamental / TAISER TADEU TEIXEIRA
BARROS. -- 2020.
174 f.
Orientador: ELISEO BERNI REATEGUI.

Coorientador: ADRIANO CANABARRO TEIXEIRA.

Tese (Doutorado) -- Universidade Federal do Rio
Grande do Sul, Centro de Estudos Interdisciplinares em
Novas Tecnologias na Educação, Programa de
Pós-Graduação em Informática na Educação, Porto
Alegre, BR-RS, 2020.

1. Pensamento Computacional. 2. Formação de
Professores. 3. Scratch. I. REATEGUI, ELISEO BERNI,
orient. II. TEIXEIRA, ADRIANO CANABARRO, coorient.
III. Título.



SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
CENTRO INTERDISCIPLINAR DE NOVAS TECNOLOGIAS NA EDUCAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA NA EDUCAÇÃO

**ATA SOBRE A DEFESA DE TESE DE DOUTORADO
TAISER TADEU TEIXEIRA BARROS**

Às nove horas do dia vinte de novembro de dois mil e vinte, no endereço eletrônico <https://mconf.ufrgs.br/webconf/00126356>, conforme a portaria 2291 de 17/03/2020 que suspende todas as atividades presenciais possíveis, nesta Universidade, reuniu-se a Comissão de Avaliação, composta pelos Professores Doutores: Crediné Silva de Menezes, Christian Brackmann, Carla Beatris Valentini para a análise da Defesa de Tese de Doutorado intitulada “**Formação em Pensamento Computacional utilizando Scratch para Professores de Matemática e Informática da Educação Básica**”, do doutorando de Pós – Graduação em Informática na Educação Taiser Tadeu Teixeira Barros sob a orientação do Prof. Eliseo Berni Reategui e coorientação do Prof. Adriano Canabarro Teixeira.

A Banca, reunida, após a apresentação e arguição, emite o parecer abaixo assinalado.

Considera a Tese Aprovada
() sem alterações;
() sem alterações, com voto de louvor;
(x) e recomenda que sejam efetuadas as reformulações e atendidas as sugestões contidas nos pareceres individuais dos membros da Banca;

Considera a Tese Reprovada.

Considerações adicionais (a critério da Banca):

Destaca-se a relevância da tese para a formação de professores para o Pensamento Computacional. A pesquisa está bem construída e apresenta resultados consistentes respondendo ao problema proposto.

Prof. Dr. Eliseo Berni Reategui
Orientador

Prof. Dr. Adriano Canabarro Teixeira
Coorientador

_____(videoconferência)_____
Prof. Dr. Crediné Silva de Menezes
PPGIE/ UFRGS

_____(videoconferência)_____
Prof. Dr. Christian Brackmann
IFFAR

_____(videoconferência)_____
Prof.^a Dr.^a Carla Beatris Valentini
UCS

Dedico este trabalho aos meus pais, esposa e filhos, amo vocês!

Agradecimentos

Agradeço primeiramente aos meus filhos e à minha esposa por me darem apoio incondicional, principalmente nos momentos em que precisei me ausentar do convívio com eles para poder realizar este trabalho! E não poderia deixar de agradecer especialmente ao meu filho Matheus, pelas opiniões sobre os programas *Scratch* e por me ajudar na utilização do *Dr. Scratch* copiando *links* e salvando pontuações.

Aos meus pais por sempre prezarem por minha educação!

Ao meu orientador professor Dr. Eliseo Berni Reategui por ter oferecido sua preciosa orientação e sempre estar disposto a me ajudar na condução deste trabalho!

Ao meu coorientador professor Dr. Adriano Canabarro Teixeira pelas contribuições teóricas e sugestões de modificação do texto!

Aos estimados professores que foram sujeitos de minha pesquisa e dos quais tive o privilégio de acompanhar a ação docente.

A Secretaria de Educação e aos profissionais do município onde a pesquisa foi realizada, que cederam seu tempo para permitir a realização deste trabalho!

Ao SENAI RS pela disponibilização de laboratórios e apoio à este trabalho!

Ao meu colega do PPGIE Ricardo Radaelli pela parceria e disponibilidade para contribuir com a etapa de codificação! E aos demais colegas pela parceria e troca de conhecimentos.

A Universidade Federal do Rio Grande Do Sul, ao PPGIE e todos os seus professores e funcionários por proporcionarem formação de tamanha qualidade.

“That which does not kill us makes us stronger”. Friedrich Nietzsche

Resumo

A formação de professores que estão em pré-serviço, assim como a formação daqueles que já atuam profissionalmente são de extrema importância para o sistema educacional. O docente, durante sua vida profissional busca estar atualizando constantemente seus conhecimentos para que sua atuação tenha impacto significativo sobre seus alunos. No atual contexto tecnológico, em que a informação está disponível para todos, o papel do professor se torna ainda mais essencial como mediador do conhecimento. Dentre a infinidade de saberes essenciais para o cidadão do novo milênio, destaca-se o Pensamento Computacional, que é considerado por alguns autores como sendo uma habilidade tão importante quanto aptidões fundamentais como a leitura e a escrita. Esta tese teve como objetivo compreender como professores de Matemática e Informática dos anos finais do ensino fundamental se apropriaram dos conhecimentos de um curso de formação em Pensamento Computacional aplicando estes conceitos em atividades de sala de aula. A metodologia foi estruturada como um estudo de caso de caráter descritivo, com uma análise de dados quali-quantitativa. No decorrer da metodologia, foram realizadas duas etapas principais: (1) o curso de formação em Pensamento Computacional para 49 professores dos anos finais do ensino fundamental e (2) atividades realizadas nas escolas onde aqueles profissionais atuavam. O curso de formação foi realizado entre setembro de 2017 e outubro de 2018, e as atividades nas escolas foram realizadas entre abril e dezembro de 2019. Em cada uma das etapas citadas, foram gerados dados para análise. Os dados gerados foram provenientes de entrevistas com os professores e de programas criados por eles, utilizando conceitos do Pensamento Computacional. As ferramentas utilizadas para analisar os dados foram a análise de conteúdo das falas dos professores e o *Dr. Scratch* para pontuar os programas que os professores criaram no *Scratch*. Ao analisar os dados coletados observou-se que o trabalho interdisciplinar entre a Matemática e a Informática geraram melhores resultados em relação à atuação individual de um docente de uma destas disciplinas isoladamente. Dentre os resultados obtidos na tese destacaram-se a certificação de 49 professores após a etapa de formação com um total de 50 horas de atividade realizadas e produção de 115 programas no *Scratch*. E na etapa de atividades em 7 escolas visitadas foram realizadas 25 horas de atividades, onde professores e alunos produziram 56 programas no *Scratch*. A análise quantitativa dos dados referentes aos programas criados evidenciou que os alunos com melhor desempenho nas atividades foram aqueles que receberam orientação dos professores que trabalharam de forma interdisciplinar. E a forma como os alunos desenvolveram seus programas foi influenciada diretamente pelos professores, com os conceitos do Pensamento Computacional com maior pontuação para os alunos possuindo uma forte correlação com os conceitos mais pontuados para os professores.

Palavras-chaves: Pensamento Computacional. Formação de Professores. Análise de Conteúdo. *Scratch*.

Abstract

BARROS, TAISER TADEU TEIXEIRA. *Computational Thinking formation using Scratch for Mathematics and Informatics Teachers from Elementary Education*. Tese (Doutorado) – Universidade Federal do Rio Grande do Sul, Centro de Estudos Interdisciplinares em Novas Tecnologias na Educação, Programa de Pós-Graduação em Informática na Educação, Porto Alegre, BR-RS, 2020.

The training of teachers who are in pre-service, as well as the training of those who already work professionally are extremely important for the educational system. The teacher, during his professional life seeks to be constantly updating his knowledge once it has significant impact on your students. In the current technological context, in which information is available to everyone, the role of the teacher becomes even more essential as a mediator of knowledge. Among the infinity of essential knowledge for the citizen of the new millennium, Computational, which is considered by some authors to be a skill as important as fundamental skills such as reading and writing. This thesis aimed to understand how mathematics and informatics teachers in the final years of elementary school have appropriated the knowledge of a training course in Computational Thinking applying these concepts in classroom activities. The methodology was structured as a descriptive case study, with a quali-quantitative data analysis. During the methodology, two main stages were carried out: (1) the Computational Thinking training course for 49 teachers in the final years of elementary school and (2) activities carried out in the schools where those professionals worked. The training course was held between September 2017 and October 2018, and activities in schools were carried out between April and December 2019. In each of the steps mentioned, data were generated for analysis. The data generated came from interviews with teachers and programs created by them, using Computational Thinking concepts. The tools used to analyzing the data were the content analysis of the teachers' statements and use of *Dr. Scratch* to score the programs that teachers created with *Scratch*. When analyzing the data collected, it was observed that the interdisciplinary work between Mathematics and Informatics generated better results in relation to the individual performance of a teacher in one of these disciplines in isolation. Some results obtained in this thesis included certification of 49 teachers after the teacher training stage with a total of 50 hours of activities carried out and production of 115 programs in the *Scratch*. And in the activities stage in 7 schools visited, 25 hours of activities were carried out, where teachers and students produced 56 *Scratch* programs. The quantitative analysis of the data referring to the created programs showed that the students with better performance in the activities were those that received guidance from teachers who worked in an interdisciplinary way. And the way in which students developed their programs was directly influenced by the teachers. The high score for concepts of Computational Thinking from students had corresponding with the high scored concepts for teachers which demonstrates a strong correlation explained by teachers' interaction with students.

Key-words: Computational Thinking. Teacher Training. Content Analysis. *Scratch*.

Lista de ilustrações

Figura 1 – Saída esperada para o exercício de algoritmos.	45
Figura 2 – IDE do <i>Scratch</i>	48
Figura 3 – Relatório gerado pelo <i>Dr. Scratch</i>	51
Figura 4 – Metodologia da tese.	69
Figura 5 – Função $y = x^2 + 2$ implementada no <i>Scratch</i>	74
Figura 6 – Estrutura condicional “SE - ENTÃO” no <i>Scratch</i>	75
Figura 7 – Sincronização implementada no <i>Scratch</i>	75
Figura 8 – Paralelismo implementado no <i>Scratch</i>	76
Figura 9 – Algoritmo implementado no <i>Scratch</i>	76
Figura 10 – Programa exemplo com comando de movimento implementado no <i>Scratch</i>	79
Figura 11 – Programa exemplo para discussão sobre o conceito de raciocínio lógico.	80
Figura 12 – Pontuação do <i>Dr. Scratch</i> para o programa da figura 11.	81
Figura 13 – Número de acessos às atividades.	82
Figura 14 – Turma do turno da manhã desenvolvendo atividades.	91
Figura 15 – Quantidade de Ocorrências dos Códigos.	95
Figura 16 – Apresentação de relatos dos professores.	104
Figura 17 – Pontuação total dos professores para o primeiro e o terceiro programa.	107
Figura 18 – Chaveiro produzido na escola BJ para os participantes da oficina.	116
Figura 19 – Projetos <i>Scratch</i> das professoras na escola BJ.	117
Figura 20 – Informações dos Projetos criados na escola BJ.	118
Figura 21 – Ação do pesquisador e professora com alunas na escola BJ.	118
Figura 22 – Ações com os alunos na escola DL.	120
Figura 23 – Ações na escola LS.	122
Figura 24 – Interação do pesquisador com os alunos na escola LS.	122
Figura 25 – Mostra de trabalhos na escola ST destacando o <i>Scratch</i>	123
Figura 26 – Programas criados na escola ST.	124
Figura 27 – Programa sobre equação do segundo grau.	126
Figura 28 – Interação do pesquisador com os alunos.	126
Figura 29 – Mostra de trabalhos.	127
Figura 30 – Quantidade de ocorrências dos códigos pré/pós visitação nas escolas.	129

Lista de quadros

Quadro 1 – Categorias e Códigos.	93
--	----

Lista de tabelas

Tabela 1 – Dimensões do Pensamento Computacional.	50
Tabela 2 – Consistência interna do questionário conforme valores de α	85
Tabela 3 – Resultados do teste t para as duas aplicações do questionário.	86
Tabela 4 – Número de professores em relação à entrega de projetos do <i>Scratch</i>	97
Tabela 5 – Pontuação do <i>Dr. Scratch</i> para o primeiro e o terceiro programa.	107
Tabela 6 – Pontuação média das dimensões do Pensamento Computacional.	108
Tabela 7 – Datas de visitas e atividades realizadas nas escolas.	113
Tabela 8 – Comparação de ocorrências dos códigos pré/pós visitação nas escolas.	133
Tabela 9 – Pontuação do <i>Dr. Scratch</i> para programas criados pelos professores e alunos.	135

Lista de abreviaturas e siglas

AVA	Ambiente Virtual de Aprendizagem
BNCC	Base Nacional Comum Curricular
CC	Ciência da Computação
CFP	Centro de Formação Profissional
CNC	Comando Numérico Computadorizado
CNE	Conselho Nacional da Educação
CP	Comunidade de Prática
CPU	<i>Central Processing Unit</i>
EMEF	Escola Municipal de Ensino Fundamental
ENIAC	<i>Electronic Numerical Integrator and Computer</i>
EUA	Estados Unidos da América
IA	Inteligência Artificial
IAS	Instituto Ayrton Senna
IoT	<i>Internet of Things</i>
IBGE	Instituto Brasileiro de Geografia e Estatística
IDE	<i>Integrated Development Environment</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
LCC	Licenciatura em Ciência da Computação
LDB	Lei de Diretrizes e Bases da Educação Nacional
MEC	Ministério da Educação
MEPeCoC	Metodologia para Ensino de Pensamento Computacional para Crianças
MIT	<i>Massachusetts Institute of Technology</i>
UFRGS	Universidade Federal do Rio Grande do Sul

UNED	<i>Universidad Nacional de Educación a Distancia</i>
PC	Pensamento Computacional
PcD	Pessoa com Deficiência
PPGIE	Programa de Pós-Graduação em Informática na Educação
RA	Realidade Aumentada
SENAI	Serviço Nacional de Aprendizagem Industrial
TDIC	Tecnologias Digitais da Informação e da Comunicação
TI	Tecnologia da Informação
TIC	Tecnologias da Informação e Comunicação

Sumário

1	Introdução	26
2	Pensamento Computacional	33
2.1	As Bases do Pensamento Computacional	38
2.1.1	Decomposição	38
2.1.2	Reconhecimento de Padrões	39
2.1.3	Abstração	39
2.1.4	Algoritmos	40
2.2	Conceitos do Pensamento Computacional	41
2.3	Atividades Plugadas X Atividades Desplugadas	42
2.3.1	Atividades Plugadas	43
2.3.2	Atividades Desplugadas	44
2.4	<i>Scratch</i>	46
2.5	<i>Dr. Scratch</i>	49
3	Formação de Professores	53
3.1	Formação de Professores em Relação ao Pensamento Computacional	60
3.1.1	Trabalhos relacionados	62
4	Metodologia	68
4.1	O curso de formação	70
4.1.1	Primeiro encontro presencial em 22/09/2017	77
4.1.2	Segundo encontro presencial em 20/10/2017	83
4.1.2.1	Questionário sobre intenção de utilização do computador	84
4.1.2.2	Entrevista	85
4.1.3	Terceiro encontro presencial em 02/03/2018 e 16/03/2018	87
4.1.4	Quarto encontro presencial em 04/05/2018 e 11/05/2018	91
4.1.4.1	Entrevistas	91
4.1.4.2	Análise de Conteúdo	92
4.1.4.3	Considerações	100
4.1.5	Quinto encontro presencial em 06/07/2018 e 13/07/2018	102
4.1.6	Sexto encontro presencial 22/10/2018	103
4.1.7	Análise dos Programas	104
4.2	Visitação e Atividades nas Escolas	113
4.2.1	Escola GH	114
4.2.2	Escola BJ	114

4.2.3	Escola DL	118
4.2.4	Escola LS	121
4.2.5	Escola ST	122
4.2.6	Escola JR	125
4.2.7	Escola EK	126
4.3	Análise de Conteúdo	127
4.4	Análise dos Programas	135
5	Conclusões e Trabalhos Futuros	142
	Referências	147
	Apêndices	161
APÊNDICE A	Termo de Consentimento	162
APÊNDICE B	Certificado	165
APÊNDICE C	Questionário sobre intenção de utilização do computador	166
APÊNDICE D	Lista dos programas entregues pelos professores durante a formação	167
APÊNDICE E	Pontuação dos programas entregues pelos professores durante a formação	169
APÊNDICE F	Pontuação do <i>Dr. Scratch</i> dos professores e alunos na etapa de visitação às escolas	171

Trajetória Acadêmica e Profissional do Autor

Concluí minha graduação em Engenharia Elétrica no ano de 2003 e logo inciei minha atuação na indústria como engenheiro de produto, permanecendo nesta área até o ano de 2008. Durante estes cinco anos avancei em meus estudos com uma pós graduação em Engenharia de Produção e Manufatura. E durante este período de cinco anos tive também um breve contato com a docência ministrando aulas de inglês.

Em 2008 surgiu a oportunidade de atuar como instrutor de nível técnico no SENAI e então comecei a me dedicar à docência, com a oportunidade de participar de uma formação pedagógica para docentes técnicos. Como passei a residir mais próximo à Porto Alegre decidi também complementar meus estudos ingressando no mestrado na UFRGS, ainda com foco na área técnica, seguindo a linha de pesquisa da robótica.

Em consonância com o mestrado, decidi investir na área da robótica educacional, fundando a *Smart Robotics*, empresa onde atuo ministrando cursos de robótica educacional, temática *Maker* e prestando consultoria na área de automação. Logo após ingressar no mestrado comecei também a atuar como docente nos cursos superiores de Ciência e Engenharia da computação.

Já pensava em seguir meus estudos após a conclusão do mestrado, porém, após a conclusão do mesmo acabei permanecendo “de molho” por aproximadamente um ano sem saber ao certo se seguiria meus estudos na mesma área de pesquisa do mestrado.

E uma vez que os cursos de robótica educacional possuem forte relação com a programação e a lógica envolvida, comecei a me interessar pelo tema, fato este que me conduziu à pesquisa sobre o Pensamento Computacional (PC). Quando ingressei no PPGIE da UFRGS, estava interessado particularmente em trabalhar com um tema relacionado à robótica educacional.

E já que estava constantemente em contato com o tema devido à atuação de minha empresa, junto ao meu professor orientador decidimos por focar a pesquisa no Pensamento Computacional em específico na questão da formação de professores, dada a importância e urgência do tema.

1 Introdução

A capacidade criativa do ser humano pode ser avaliada pelos inventos nas mais diversas áreas do conhecimento, tais como as Artes, Ciências e Engenharias. A invenção da roda, em 3.500 a.C (BONDAR, 2018), foi uma das obras mais importantes da Era do Cobre.

A roda possibilitou uma revolução nos meios de transportes sendo utilizada como acessório desde as antigas charretes, movidas a tração animal, passando pelas locomotivas movidas a vapor, e, atualmente, equipando nossos automóveis, tanto aqueles acionados a combustão, como os elétricos.

Com o intuito de interpretar a natureza e seus fenômenos, o ser humano deu “vida” à Matemática, que permitiu abstrair os fenômenos naturais descrevendo-os sob a forma de equações. Outras ciências, como a Física, a Química e a Mecânica, também foram desenvolvidas para servir como ferramentas intelectuais. Estes instrumentos moldaram a tecnologia por permitir que novos utensílios, fórmulas, processos e máquinas fossem projetados e criados.

Os autômatos foram uma tentativa humana de criar máquinas à imagem e semelhança de sua inteligência. Alguns destes dispositivos autômatos encantavam plateias, assim como foi o caso do famoso autômato¹ jogador de xadrez “o Turco”, do Século XVIII.

Mas nenhuma outra invenção foi tão bem sucedida na tarefa de “imitar” a inteligência humana quanto o computador. A norma 1680.1-2009, do Instituto de Engenheiros Elétricos e Eletrônicos (IEEE²), define o computador como um dispositivo que executa operações lógicas e processa dados. O computador deve ser composto por, no mínimo: (a) uma CPU (*Central Processing Unit*) que realize operações; (b) um dispositivo de entrada de usuário como o teclado; e (c) um dispositivo de exibição de informação como um monitor³.

Para evoluir desde os primeiros computadores “rudimentares”, até os modelos atuais, foram necessários muitos estudiosos e inventores como foi o caso de *Charles Babbage*, que, em 1832, apresentou sua máquina de diferenças⁴. A máquina de *Babbage* era um computador mecânico ao qual somente o próprio inventor *Babbage* tinha acesso e conhecimento para operá-lo. Posteriormente, a máquina de *Babbage* foi estudada por *Ada Lovelace*, que produziu aquele que foi considerado por muitos como o primeiro programa de computador.

¹ <https://bit.ly/2AbpJBy>, acessado em 29/12/2018.

² Minha tradução para *Institute of Electrical and Electronics Engineers*, acessado em 23/10/2019.

³ Este dispositivo pode também ser um dispositivo de áudio no caso de pessoa com deficiência visual.

⁴ <https://www.britannica.com/biography/Charles-Babbage>, acessado em 25/10/2019.

Após 123 anos da invenção da máquina de *Babbage*, foi apresentado ao mundo o Integrador Eletrônico Numérico e Computador⁵ (ENIAC). O ENIAC era utilizado exclusivamente para fins militares e por algumas universidades. Os programadores do ENIAC além de possuírem acesso restrito ao equipamento, eram pessoas dotadas de pleno conhecimento técnico sobre o mesmo. Somente na década de 70 que os primeiros computadores pessoais começaram a ganhar o mercado, como foi o caso⁶ dos computadores da *Apple* e da *Commodore*.

No entanto, as primeiras gerações de computadores pessoais ainda contavam com um público seletivo pois a operação daqueles exigia conhecimentos específicos por parte dos usuários, inclusive fundamentos de programação.

Desde as primeiras gerações de computadores pessoais, que surgiram na década de 70, até os dias atuais, muitas marcas e modelos foram criados. Além de haver maior quantidade de modelos disponíveis para uso pessoal, o acesso a estes equipamentos foi ampliado consideravelmente, uma vez que o custo foi sendo reduzido.

O número de modelos de computadores disponíveis, com o passar dos anos, foi ficando cada vez maior e, atualmente, o computador se apresenta com uma diversidade de formatos e configurações.

Podemos adquirir um computador para ocupar uma posição fixa em nossa mesa (*Desktop*) ou para nos acompanhar em diferentes atividades (*Notebook*). Ainda, um dispositivo que já serviu somente para realizar ligações telefônicas, atualmente possui todas as funcionalidades do computador (*Smartphone*, sendo praticamente uma extensão do nosso corpo).

No Brasil, conforme pesquisa⁷ do Instituto Brasileiro de Geografia e Estatística (IBGE), realizada em 2016, o *Smartphone* estava presente em mais de 64 milhões de residências visitadas. Números como este, mostram que o computador é um equipamento cada vez mais acessível à população e atualmente o uso do *Smartphone* como uma ferramenta de sala de aula e de acesso as atividades em um AVA (Ambiente Virtual de Aprendizagem) é uma realidade.

Porém, a popularização do computador parece ser inversamente proporcional à sua utilização como um instrumento científico. Ou seja, emprega-se o computador mais como elemento de lazer e diversão.

Do computador mecânico e rudimentar, que era a máquina de *Babbage*, passando pelo ENIAC, e chegando nas primeiras gerações de computadores pessoais da década de 70, todos eram equipamentos que tinham propósitos específicos. Geralmente precisavam ser

⁵ Minha tradução para *Electronic Numerical Integrator and Computer*, <https://www.britannica.com/technology/ENIAC>, acessado em 30/10/2019.

⁶ <https://www.ithistory.org/hardware?page=20>, acessado em 10/10/2019.

⁷ Dados disponíveis em <https://agenciadenoticias.ibge.gov.br>, acessado em 30/12/2018.

reprogramados para cada nova tarefa, exigindo de seus usuários conhecimentos específicos do *hardware* do computador e de como programá-lo, isto é, do *software* daquele.

A necessidade de conhecer o *hardware* e o *software* dos computadores foi diminuindo com o passar dos anos. Atualmente, o usuário do computador pode acessar uma rede social, assistir um vídeo ou jogar sem precisar entender como o computador funciona, ou mesmo escrever uma única linha de código.

Porém, em uma sociedade na qual o computador está cada vez mais presente, e assume tarefas humanas, é essencial compreender como o mesmo funciona. Assim, as tecnologias digitais, a Inteligência Artificial (IA) e a robótica, dentre outras tecnologias, passam obrigatoriamente pelo uso do computador.

Uma das formas de compreender o computador, o qual em alguns aspectos mime-tiza a inteligência humana, é compreendendo o Pensamento Computacional (PC), o qual caracteriza-se como um conjunto de habilidades que podem ser identificadas diretamente na área computacional mas que também podem ser generalizadas e aplicadas em outras áreas do conhecimento. Dominar os conceitos relativos ao Pensamento Computacional permite que o indivíduo resolva problemas de uma forma mais sistematizada e eficiente.

Compreender o Pensamento Computacional vai além de compreender o funcionamento de um computador ou saber programar, pois, mesmo sem um computador, podemos desenvolver atividades do Pensamento Computacional. Ainda, a própria concepção do computador utilizou os conceitos do Pensamento Computacional.

A utilização de conceitos computacionais aplicados em outras áreas do conhecimento foi apresentada por [Ferreira et al. \(2015\)](#) trazendo exemplos nas áreas da Matemática, Biologia, Artes e Língua Portuguesa durante a realização de atividades com um grupo de alunos. Na área da Matemática, foi citada a utilização de busca sequencial e busca binária, exemplificando como procurar valores dentro de um conjunto numérico. Na Biologia, os alunos seguiram uma sequência de etapas (algoritmo) para resolver um problema de desequilíbrio ecológico. Além disso, o algoritmo foi utilizado para criar uma coreografia para a dança nas Artes. E na Língua Portuguesa, por sua vez, os alunos aprenderam como reduzir o tamanho de um texto eliminando palavras repetidas relacionando esta tarefa com a compactação de arquivos. Os autores demonstraram a utilização de atividades relacionadas ao Pensamento Computacional sem necessariamente utilizar um computador e apresentaram⁸ a relação entre o chamado raciocínio computacional e o Pensamento Computacional: “O raciocínio computacional é utilizado de forma mais específica, quando o pensamento computacional está relacionado ao pensamento analítico e ao raciocínio dedutivo que envolve a lógica e a matemática”.

No atual contexto tecnológico, o Pensamento Computacional serve como base de

⁸ Ibidem. p. 257.

conhecimento que permite aos interessados adentrar em outros campos de estudo como os *Fablabs*, que são espaços compartilhados onde se pode projetar e fabricar artefatos usando, por exemplo, cortadoras *laser* e impressoras 3D.

Além do mais, nos *Fablabs*, os usuários são encorajados a documentar e compartilhar abertamente seus projetos (KOHTALA, 2016).

O contexto tecnológico dos *Fablabs* é o mesmo da cultura do “Faça Você Mesmo” do inglês “Do It Yourself (DIY)”, que surgiu graças ao acesso facilitado à informação tecnológica para qualquer pessoa.

Por exemplo: as máquinas de Controle Numérico Computadorizado (CNC) que há algumas décadas eram de uso exclusivo de empresas de grande porte, hoje podem ser adquiridas em *sites* especializados por valores acessíveis às empresas de pequeno porte e até mesmo para *hobbistas*.

As máquinas CNCs poderiam ser utilizadas, por exemplo, por um artesão que necessitasse recortar peças em madeira e incrementar sua produção. Mas se fosse inviável financeiramente para este artesão adquirir uma máquina CNC? A resposta para esta pergunta veio com a cultura DIY, pois surgiram uma diversidade de *sites* especializados explicando como produzir uma CNC, utilizando até mesmo sucata, que pode ser adquirida em um ferro velho por valores baixos. Assim seria possível para o artesão obter uma máquina CNC viável financeiramente e que atendesse sua produção artesanal.

Contudo, em algum momento, quando o artesão fabricasse sua própria máquina CNC, necessitaria entrar em contato com um processo computacional relacionado à gravação de um *firmware*, ou ao *setup* e comissionamento de parâmetros da máquina, ou até mesmo a necessidade de criar um arquivo para impressão 3D. Logo, caso houvesse uma relação direta com o Pensamento Computacional, todas estas atividades descritas seriam otimizadas.

Na literatura, a discussão sobre o Pensamento Computacional é vasta e enfatiza, dentre outros, que dominar o Pensamento Computacional pode trazer benefícios com relação a outros campos do conhecimento. Como um exemplo de discussão apresentada, Mohaghegh e Mccauley (2016) trazem que a inovação⁹ é uma característica chave do Pensamento Computacional.

⁹ Para M.Z et al. (2016) a inovação se refere a qualquer prática nova para as organizações, incluindo equipamentos, produtos, serviços, processos, políticas e projetos. Sendo esta definição similar no contexto educacional conforme proposto por Vincent-Lancrin et al. (2019, p. 21): “um produto ou processo novo ou aprimorado (ou combinação dos mesmos) que difere significativamente dos produtos ou processos anteriores da unidade e que foram disponibilizados para usuários em potencial (produto) ou usado pela unidade (processo)”. Minha tradução para o texto original: “*a new or improved product or process (or combination thereof) that differs significantly from the unit’s previous products or processes and that has been made available to potential users (product) or brought into use by the unit (process)*”.

Os autores¹⁰ citaram que pensar de forma inovadora faz com que os estudantes questionem coisas já existentes, desafiem suposições e pensem “fora da caixa”, sendo que este aspecto traz para os chamados “Pensadores Computacionais” uma vantagem significativa na resolução de problemas.

Outro tópico discutido por [Mohagheh e Mccauley \(2016\)](#) foi a relação do Pensamento Computacional com a educação, uma vez que se tornou cada vez mais importante compreender o que pode e o que não pode ser resolvido por processos computacionais.

Compreender os processos computacionais permite aos estudantes buscar/conceber modelos computacionais para situações que tradicionalmente não estão relacionadas à Ciência da Computação. E mesmo que um aluno opte por uma carreira diferente da computação, as habilidades aprendidas e desenvolvidas, através do Pensamento Computacional, podem ser utilizadas em qualquer outro campo do conhecimento.

O desenvolvimento de atividades do Pensamento Computacional, em sala de aula, é uma realidade em diversos países e instituições, sendo um assunto discutido por diversos autores ([BOWER et al., 2017](#); [MIOTTO](#); [CARDOSO, 2014](#); [OLIVEIRA, 2009](#); [BRACKMANN, 2017](#); [MORAIS; BASSO; FAGUNDES, 2017](#)) que propuseram diferentes modelos para implementação de práticas voltadas para o desenvolvimento do Pensamento Computacional nas escolas. Os modelos que foram propostos trouxeram desde a criação de disciplinas específicas, até a realização de atividades extra-curriculares relacionadas ao Pensamento Computacional.

No Brasil, as discussões sobre a inclusão das atividades relativas ao Pensamento Computacional intensificaram-se, em grande parte, devido à inclusão do Pensamento Computacional nas diretrizes da Base Nacional Comum Curricular¹¹ (BNCC).

Mas, analisando-se a BNCC, foi possível observar que o documento não trouxe indicações de como o Pensamento Computacional poderia ser integrado ao currículo escolar, ou qual tipo de formação complementar deveria ser oferecida aos professores envolvidos. Assim, esta tese trouxe uma contribuição como material disponível para formação dos professores alinhado às necessidades apresentadas na BNCC.

A BNCC trouxe a maior parte das citações ao termo pensamento computacional referenciadas na área da Matemática. Porém, de uma forma geral, na literatura pesquisada, as áreas que foram citadas para trabalhar com o Pensamento Computacional também incluem a Informática.

[Leite et al. \(2017\)](#) afirmaram que os cursos de Licenciatura em Informática têm a responsabilidade de formar professores para disseminar o Pensamento Computacional, trazendo uma ideia diferenciada com relação à BNCC, que imputou esta responsabilidade

¹⁰ Ibidem.

¹¹ Documento normativo que define o conjunto de aprendizagens essenciais que os alunos devem desenvolver ao longo da Educação Básica

aos professores da Matemática.

Assim como a ideia que foi apresentada na BNCC, existe uma relação direta do Pensamento Computacional com a Matemática, bem como foi relatado por [Barcelos e Silveira \(2013\)](#), que identificaram três grupos de competências comuns ao Pensamento Computacional e à Matemática:

[...] alternar a representação de problemas e suas soluções entre diferentes representações semióticas (verbal, algorítmica, matemática); identificar regularidades e padrões; elaborar e interpretar modelos que descrevem e representam problemas a serem resolvidos.

Parece evidente que existe a necessidade da introdução dos conceitos do Pensamento Computacional no currículo escolar e que não existe uma definição exata sobre qual a formação dos professores responsáveis por esta tarefa.

De uma forma geral, os professores apresentam resistência ao uso de tecnologias, tal como é o caso da utilização do computador em sala de aula, demonstrando insegurança principalmente pelo fato de que não tiveram formação apropriada enquanto se preparavam para a carreira docente ([TALLVID, 2016](#); [SCHUHMACHER; FILHO; SCHUHMACHER, 2017](#); [ZANELLA; LIMA, 2017](#)).

Dada a importância da introdução do Pensamento Computacional no currículo escolar, e da necessidade de formação de professores capazes de ministrar tópicos do Pensamento Computacional, definiu-se a seguinte questão norteadora da pesquisa:

“Em um contexto de formação de professores do ensino fundamental para o desenvolvimento do pensamento computacional, de que maneira estes educadores se apropriam e mobilizam os conhecimentos trabalhados no curso em atividades práticas de sala de aula?”.

Após definida a questão norteadora da pesquisa, também foi definido o **Objetivo Geral** desta tese: *Compreender como professores de Matemática e Informática, dos Anos Finais do ensino fundamental, se apropriam dos conhecimentos de um curso de formação em Pensamento Computacional, aplicando estes em atividades de sala de aula.*

Também, os seguintes objetivos específicos foram definidos:

- Avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional/*Scratch* durante a formação;
- Avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional/*Scratch in loco* nas escolas onde atuam, após a etapa de formação;

- Identificar as diferenças na percepção sobre a formação e nas atividades realizadas em sala de aula de acordo com a formação do professor (Informática/Matemática);

Para alcançar estes objetivos, estruturou-se um estudo com base em um curso de formação de professores para trabalhar com o Pensamento Computacional. Conforme a natureza do estudo realizado classificou-se o mesmo como um estudo de caso. Uma definição de estudo de caso proposta por Wohlin et al. (2012) traz que este tipo de estudo investiga uma única entidade ou fenômeno, onde o pesquisador coleta informações detalhadas durante um período de tempo determinado utilizando variados procedimentos e perspectivas de análise.

O curso de formação que foi elaborado teve uma carga horária de 50 horas-aula. As aulas aconteceram no período entre 22 de Setembro de 2017 e 22 de Outubro de 2018. Durante este período foram coletados dados sobre o desempenho dos professores nas atividades propostas. Também foram realizadas entrevistas para conhecer a percepção dos professores sobre o curso de formação. Esta etapa permitiu atender o objetivo específico de avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional e do *Scratch*.

Posteriormente aconteceu uma etapa de visitação e atividades nas escolas no período entre 22 de Abril e 09 de Dezembro de 2019, com a presença de alguns dos professores que participaram da formação, bem como, com a presença de alguns alunos. Nesta etapa novamente foram realizadas entrevistas com os professores e adicionalmente foram realizadas análises de programas criados por professores e alunos atendendo o objetivo específico de avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional e do *Scratch in loco* nas escolas.

Os dados gerados nas duas etapas citadas foram analisados e permitiram atender o objetivo específico de identificar as diferenças na percepção sobre a formação e nas atividades realizadas em sala de aula de acordo com a formação do professor.

A estrutura da tese foi organizada da seguinte forma: o Capítulo 1 introdutório contextualizou a tese, seguido pelo Capítulo 2, que trouxe uma revisão da literatura referente à temática do Pensamento Computacional, discutindo também as ferramentas *Scratch* e *Dr. Scratch*.

O Capítulo 3 revisa a literatura no âmbito da formação de professores e trabalhos relacionados com a formação de professores especificamente relacionada ao Pensamento Computacional.

O Capítulo 4 apresenta a etapa Metodológica, descreve o curso de formação e visitas realizadas nas escolas, traz os resultados das análises de conteúdo das entrevistas com professores e os programas de professores e alunos avaliados no *Dr. Scratch*. Por fim, a última seção apresenta conclusões e trabalhos futuros.

2 Pensamento Computacional

Introdução

O Pensamento Computacional (PC) pode ser compreendido como uma forma de utilizar o computador não apenas como uma simples ferramenta para produzir textos ou acessar a *World Wide Web*, mas também como uma ferramenta poderosa que pode auxiliar o homem na resolução de problemas.

Além disso, o Pensamento Computacional transcende e amplia a definição de ser um conhecimento relacionado somente ao uso do computador, uma vez que pode ser utilizado nos mais diversos campos do conhecimento humano.

Foi Seymour Papert (1980, p. 182) no livro *Mindstorms: Children, Computers, and Powerful Ideas* que utilizou o termo Pensamento Computacional pela primeira vez: “Suas visões de como integrar o pensamento computacional na vida cotidiana não foram suficientemente desenvolvidas. Mas haverá mais tentativas e mais e mais¹”.

O autor destacou que o aparato tecnológico da época (anos 70) limitava a compreensão da importância de introduzir o Pensamento Computacional no cotidiano educacional.

O termo Pensamento Computacional começou a ter destaque somente 26 anos após Papert ter publicado sua obra². A pesquisadora Jeannette Wing (2006), por sua vez, trouxe uma definição que representa um conjunto de atitudes e habilidades universalmente aplicáveis, em que todos - não apenas cientistas da computação - deveriam estar ansiosos para aprender e usar.

Em uma publicação posterior, Wing (2012) trouxe o Pensamento Computacional como um processo de pensamento envolvido na formulação de um problema e na expressão de sua solução de maneira que um computador, humano ou máquina pudessem realizar efetivamente.

Depois da utilização do termo Pensamento Computacional, por Wing, outros autores trouxeram novas definições assim como Paulo Blikstein (2008), que definiu o Pensamento Computacional em termos de “saber usar o computador como instrumento de aumento do poder cognitivo e operacional humano”, sendo esta, possivelmente, a habilidade mais importante a ser ensinada àqueles que necessitem estar aptos a exercer plenamente a cidadania no Século XXI.

¹ Minha tradução para o texto original: “*Their visions of how to integrate computational thinking into everyday life was insufficiently developed. But there will be more tries, and more and more.*”.

² Ibidem.

Já Araújo, Andrade e Serey (2015) contextualizaram o Pensamento Computacional como uma abordagem de resolução de problemas que incorpora processos mentais utilizando habilidades como organização, análise de dados, construção de algoritmos, abstração, criação de modelos, simulação, automatização de soluções e paralelização.

Deve-se observar, porém, que o Pensamento Computacional não está necessariamente associado ao uso do computador, uma vez que podemos, por exemplo, utilizar um algoritmo para solucionar um problema utilizando papel e caneta, ou mentalmente. Mas, obviamente, através da utilização do computador, obteríamos a execução do algoritmo, ou a resposta de um cálculo complexo, de uma forma muito mais rápida.

De uma forma similar à comparação entre a velocidade de resposta do computador com a velocidade de raciocínio do ser humano, Wing (2006) trouxe que o Pensamento Computacional retomou o enigma da inteligência de máquina: “o que nós humanos podemos fazer melhor que os computadores e o que os computadores podem fazer melhor que nós humanos”? Com a pergunta fundamental que deveria ser realizada sendo: “O que é computável”? Ou seja, o que pode ser processado por um computador de forma mais eficiente do que seria realizado pelo ser humano?

Uma brincadeira comum nos cursos de engenharia, e que os professores costumam reproduzir com os alunos, se refere ao computador como sendo um equipamento “burro”, pois apresenta rapidez de execução, comparado ao ser humano, que, por sua vez, embora detenha inteligência, apresenta lentidão ao demonstrar sua capacidade de raciocínio.

Esta brincadeira traduz, de certa forma, a importância de definirmos o que o computador deve fazer e o que nós, seres humanos, devemos. Ao ser humano compete a abstração, criação dos algoritmos e processos, que uma vez “ensinados” ao computador podem ser realizados pelo mesmo.

Pessoas “leigas” tendem a crer que os computadores são máquinas “inteligentes”, que simplesmente saem da linha de fabricação desta forma, e este tipo de concepção incorreta ocorre por uma falta de conhecimentos básicos relacionados ao Pensamento Computacional. Dentre estes conhecimentos básicos, podemos citar os algoritmos que permitiram o desenvolvimento de campos de estudo como o da Inteligência Artificial (IA) e da aprendizagem de máquina.

Não se deve esperar que todas as pessoas possam discutir e se interessar pela questão de os computadores estarem evoluindo Sterling (2016) ou questionar os rumos da IA e o que vai acontecer caso aqueles se tornem mais “espertos” que o ser humano, conforme questionado por Tim Urban (PENNSYLVANIA, 2016). Mas conhecer conceitos básicos relacionados ao Pensamento Computacional já permitiria que uma pessoa tivesse uma noção da forma como um computador processa as informações e qual a lógica envolvida. Tais condições permitem maior liberdade e autonomia para compreender diversos

acontecimentos tecnológicos que estamos vivendo.

O conhecimento tecnológico está cada vez mais inserido em todas as áreas profissionais. Atualmente, é inconcebível imaginar um cidadão ignorante em Computação (AMARAL et al., 2014), uma vez que, qualquer atividade profissional usa tecnologias da informação que estão atreladas a um raciocínio computacional (algorítmico).

Existem inúmeros problemas das Ciências Exatas, das Ciências Humanas e da realidade cotidiana que poderiam ser resolvidos utilizando o Pensamento Computacional, segundo a definição³ proposta por Csizmadia et al. (2015, p. 5):

O pensamento computacional fornece uma estrutura poderosa para o estudo da computação e sua aplicação é ampla indo além da própria computação. É o processo de reconhecer aspectos da computação no mundo que nos cerca aplicando ferramentas e técnicas da computação para entender e raciocinar sobre questões naturais, sociais, processos e sistemas artificiais. Permite que alunos resolvam problemas dividindo estes em partes menores solucionáveis, criando algoritmos para resolvê-los.

Previamente ao advento do computador, e mesmo nas primeiras décadas de utilização do mesmo, o indivíduo precisava “apenas” conhecer operações aritméticas básicas da Matemática e estar alfabetizado e letrado⁴, combinando estes saberes para poder desempenhar as tarefas do cotidiano.

Porém, no contexto tecnológico atual, cada vez mais o chamado letramento digital vai se tornando imprescindível. Assim como comenta Ribeiro (2017, p. 20): “A Internet e as máquinas digitais estão entre as opções mais recentes do letramento. Por isso uma preocupação com os usos das novas tecnologias surgiu entre aqueles que investigam leitura e escrita”.

Uma relação entre o letramento digital e o Pensamento Computacional foi proposta por Valente (2019), que também discorreu sobre os termos Letramento Computacional e Competência Digital, revelando que estas teorias convergem no sentido de auxiliar na compreensão de como podem ser implementadas educacionalmente⁵:

Os resultados das pesquisas e trabalhos sendo realizados sobre o pensamento computacional e as diferentes modalidades de letramento têm permitido o acúmulo substancial de informação sobre como as tecnologias digitais e as mídias podem ser implantadas na educação.

³ Minha tradução para o texto original: “Computational thinking provides a powerful framework for studying computing, with wide application beyond computing itself. It is the process of recognising aspects of computation in the world that surrounds us and applying tools and techniques from computing to understand and reason about natural, social and artificial systems and processes. It allows pupils to tackle problems, to break them down into solvable chunks and to devise algorithms to solve them.”

⁴ Os conceitos de alfabetização e letramento que deram apoio a este raciocínio foram os discutidos e apresentados por Goulart (2014).

⁵ Ibidem. p. 157.

Estamos vivendo um momento em que o contato com o computador e equipamentos similares, como o *Smartphone* e o *Tablet*, é cada vez mais constante e inevitável. Sendo assim, quanto tempo ainda poderá passar sem que habilidades básicas de programação e compreensão sobre o funcionamento do computador passem a ser tão fundamentais quanto a alfabetização?

O relatório do Fórum Econômico Mundial⁶ ([World Economic Forum, 2016](#)), intitulado “O futuro dos empregos - Emprego, Competências e Estratégia da força de trabalho para a Quarta Revolução Industrial”, trouxe uma perspectiva sobre como os postos de trabalho estão sendo e serão afetados pela tecnologia: tarefas repetitivas são realizadas por robôs cooperativos, atendimento a clientes são realizados por *chatbots*⁷ e dispositivos de uso comum, como os próprios eletrodomésticos, estão sendo conectados⁸ à *Internet*.

Toda esta revolução tecnológica vai acarretar o desaparecimento de alguns postos de trabalho ao mesmo tempo que vai criar outros novos, os quais, ao se apoiarem na tecnologia (enfatizando-se a tecnologia computacional), poderão favorecer os indivíduos que possuem conhecimentos associados ao Pensamento Computacional.

Migrando do mundo do trabalho para o mundo da educação, foi verificado que a necessidade de incluir o Pensamento Computacional no currículo escolar foi um assunto discutido no Brasil pelo Ministério da Educação (MEC), apresentado a relação direta da Matemática com o Pensamento Computacional no texto da BNCC (2019, p. 266–271):

Os processos matemáticos de resolução de problemas, de investigação, de desenvolvimento de projetos [...] são potencialmente ricos para o desenvolvimento de competências fundamentais para o letramento matemático [...] e para o desenvolvimento do pensamento computacional. Outro aspecto a ser considerado é que a aprendizagem de Álgebra como também aquelas relacionadas a Números, Geometria e Probabilidade e estatística, podem contribuir para o desenvolvimento do pensamento computacional [...] Associado ao pensamento computacional, cumpre salientar a importância dos algoritmos e de seus fluxogramas, que podem ser objetos de estudo nas aulas de Matemática. [...] Outra habilidade relativa à álgebra que mantém estreita relação com o pensamento computacional é a identificação de padrões para se estabelecer generalizações, propriedades e algoritmos.

A proposta da BNCC de incluir o Pensamento Computacional nos currículos escolares é fundamental para aprimorarmos o sistema educacional Brasileiro, sendo que, dentre os benefícios do ensino, e integração do Pensamento Computacional nos currículos escolares, os seguintes itens foram citados por [Lockwood e Mooney \(2017\)](#):

⁶ Organização internacional sem fins lucrativos para a Cooperação Público-Privada que envolve os principais líderes políticos e empresariais em agendas globais, regionais e setoriais empenhando-se para promover o empreendedorismo de interesse público global, mantendo altos padrões de governança. <https://www.weforum.org/>.

⁷ Robôs de *Software* capazes de simular conversas com usuários humanos.

⁸ No contexto da *IoT - Internet of Things*.

- A melhoria das habilidades analíticas dos alunos;
- A compreensão de que a programação é sobre como resolver problemas e não apenas como criar códigos;
- Melhorar atitudes e confiança das mulheres em relação à programação;
- A capacidade de prever o sucesso acadêmico, pois as pontuações do Pensamento Computacional possuem uma forte correlação com o desempenho acadêmico em geral.

A literatura internacional na área aponta para diferentes modelos de implementação do ensino do Pensamento Computacional, desde a criação de disciplinas específicas até o enquadramento do tema em práticas interdisciplinares (KONG; ABELSON, 2019).

Imberman, Sturm e Azhar (2014), no entanto, destacam que muitos dos cursos de formação oferecidos aos professores se concentram na integração do Pensamento Computacional ao currículo das disciplinas de Informática e Matemática. No contexto da formação de professores, Yadav et al. (2014) citam os professores do K-12 e discorrem sobre como deveria ser sua capacitação para trabalhar com os conceitos do Pensamento Computacional em sala de aula.

No Brasil, Valente (2016) destacou que a formação de professores para trabalhar com o Pensamento Computacional enfrenta desafios importantes por não haver um consenso sobre habilidades e conteúdos a serem trabalhados. Não obstante, ações pontuais têm sido desenvolvidas nesse sentido, como no trabalho realizado por Amaral et al. (2014). Os pesquisadores relatam experiências de ensino dos fundamentos da Computação na Educação Básica, experiências realizadas pelo curso de Licenciatura em Computação da UFPE.

Já Rocha e Prado (2014) apresentam a análise de um curso de formação continuada sobre programação em *Scratch* para professores de Matemática, levando os docentes a criar suas próprias ferramentas e refletir sobre suas práticas em sala de aula. França e Tedesco (2015), no entanto, enfatizam que a maior parte da literatura nacional sobre formação em Pensamento Computacional traz relatos de projetos de pesquisa ou de extensão, sem discutir de maneira mais contundente de que maneira o ensino do Pensamento Computacional pode ser formalizado no país.

Silva, Silva e França (2017) destacam, ainda, os desafios relacionados à inadequação da infraestrutura nas escolas para trabalhar com o desenvolvimento do Pensamento Computacional em abordagem plugada.

2.1 As Bases do Pensamento Computacional

De uma forma genérica, podemos compreender o Pensamento Computacional como um conjunto de habilidades que permitem sistematizar a resolução de problemas, pois estes podem ser resolvidos por um ser humano e adicionalmente por um computador.

Com relação às habilidades comuns ao Pensamento Computacional, foi possível identificar um grupo com quatro⁹ elementos que serão identificados nesta tese como as bases¹⁰ do Pensamento Computacional (BITESIZE, 2015; PROTTSMAN, 2019):

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

A utilização das bases do Pensamento Computacional permitem a compreensão de como o Pensamento Computacional pode ser utilizado em atividades práticas (BITESIZE, 2015). Por exemplo: quando há a necessidade em resolver um problema grande e complexo, pode-se dividir este problema em pedaços menores, e, conseqüentemente, com a menor complexidade, utiliza-se a decomposição.

Ao avaliar cada pedaço do problema maior que foi decomposto, deve-se utilizar o reconhecimento de padrões para buscar soluções parecidas para os problemas menores. Se apenas detalhes importantes forem considerados, ao resolver um dos problemas menores, ignorando informações irrelevantes para a solução, utilizar-se-á a abstração.

Por outro lado, caso seja criada uma seqüência de etapas para resolver os problemas, e que aquelas possam ser reutilizadas em problemas novos, teremos trabalhado com o conceito de algoritmos.

Cada uma das quatro bases do Pensamento Computacional pode ser utilizada individualmente, ou em conjunto com as demais, na ordem que for mais apropriada para gerar a solução para um problema. A seguir, será apresentada cada uma das bases com um maior detalhamento teórico.

2.1.1 Decomposição

A decomposição permite que um problema complexo possa ser dividido em partes menores que individualmente possuem menor complexidade e, por conseqüência, são

⁹ Há também outras contextualizações, como por exemplo a apresentada por Ribeiro, Foss e Cavaleiro (2017) que traz os “pilares do Pensamento Computacional” como sendo 3: abstração, análise e automação.

¹⁰ Minha tradução adotada para a palavra *cornerstone*.

mais fáceis de serem compreendidas e solucionadas. Definições similares a esta podem ser encontradas em [CODE \(2013\)](#) e [ISTE \(2018\)](#).

Equipes de desenvolvimento de *software* utilizam a decomposição para distribuir as tarefas aos programadores, possibilitando que um programa com milhares de linhas de código possa ser desenvolvido simultaneamente por um time composto por diversos programadores.

Um exemplo lúdico da decomposição é a montagem de um quebra-cabeça, no qual podemos dividir a montagem selecionando inicialmente as peças que compõem a borda, separando as peças por cores e, depois, por formato, facilitando, assim, a montagem.

2.1.2 Reconhecimento de Padrões

Os problemas que podem ser resolvidos computacionalmente, ou de um modo geral - utilizando o Pensamento Computacional -, possuem a característica de serem enquadrados em categorias similares que, dessa forma, também possuem soluções idênticas, sendo o papel fundamental do reconhecimento de padrões encontrar similaridades entre as coisas ([CODE, 2013](#)).

O reconhecimento de padrões é uma habilidade que envolve mapear tanto as similaridades, como as diferenças de problemas menores que foram já decompostos de um problema maior. Os indivíduos que possuem a habilidade de reconhecer padrões podem desenvolver uma base sólida na formulação de algoritmos ([ISTE, 2018](#)).

2.1.3 Abstração

A abstração é um mecanismo importante no processo de solução de um problema e permite representar os aspectos mais relevantes deste problema simplificando a realidade e propondo uma solução ([RIBEIRO; FOSS; CAVALHEIRO, 2017](#)). A abstração envolve, ainda, filtrar ou ignorar detalhes sem importância, o que essencialmente torna um problema mais fácil de entender e resolver ([ISTE, 2018](#)).

Outro aspecto fundamental da abstração é verificar diferenças específicas que permitam que uma solução única funcione para problemas distintos ([CODE, 2013](#)). Um exemplo típico da abstração é a quantificação que realizamos na Matemática, na qual abstraímos quantidades através da utilização de números que são representações pictóricas para tais quantidades.

A quantificação também exemplifica uma solução única que funciona para problemas distintos, já que podemos utilizar o número 3 para quantificar três automóveis ou três outros objetos quaisquer.

No texto da organização, [Barefoot \(2014\)](#) apresenta como exemplos de abstração:

(1) Os mapas, onde somente características específicas são mostradas como é o caso do contorno de um país para informação governamental ou dados referentes aos relatórios sobre terremotos. E (2) a descrição dos componentes de *hardware* de um computador representados como “caixas pretas”, das quais não é relevante conhecer detalhadamente cada componente que as compõem, mas, sim, sua funcionalidade.

2.1.4 Algoritmos

A palavra algoritmo pode trazer a ideia de um conceito com extrema complexidade¹¹, porém, podemos defini-lo simplesmente como uma lista de passos a serem seguidos para completar uma tarefa (CODE, 2013).

Uma receita de bolo, por exemplo, pode ser interpretada como um algoritmo simples, pois é uma sequência de passos que, quando seguidos de forma ordenada e respeitando as quantidades associadas, vai produzir um bolo delicioso.

A importância dos algoritmos reside no fato de que os mesmos conseguem unir os conhecimentos derivados das três outras bases do Pensamento Computacional em algo executável (ISTE, 2018), podendo ser interpretado e executado tanto por um humano, como por uma máquina.

Claro que, em determinados casos, assim como no exemplo de um algoritmo para ordenação numérica, a capacidade humana não permitiria classificar um conjunto contendo mais que algumas centenas de números, principalmente por restrições de tempo.

Já ao contrário do caso humano, utilizar um computador para executar um algoritmo de classificação numérica só teria limitações quanto à capacidade de memória e arquitetura do computador.

Ao comparar o ser humano com o computador, o fato fundamental é que o conhecimento necessário para criar o algoritmo, assim como a interpretação do mesmo, são capacidades humanas. A execução de um algoritmo, por sua vez, pode ser realizada tanto pelo ser humano, como pelo computador.

Uma vez conhecidas as bases do Pensamento Computacional, estas permitem que um determinado problema possa, então, ser decomposto em pedaços de menor tamanho e complexidade.

Ao criar soluções para cada pedaço do problema, podemos identificar soluções similares reconhecendo padrões e simplificando o problema pela abstração.

Ao concluir estes processos, seria possível gerar um algoritmo relacionado ao problema trazendo possíveis soluções. Mas esta ordem de utilização das bases não precisa ser

¹¹ Na minha experiência docente inclusive já presenciei em mais de uma oportunidade alunos que confundem a palavra algoritmo com logaritmos que é um conceito matemático.

seguida necessariamente.

Com relação às etapas para solucionar um problema relacionado ao Pensamento Computacional, [Repenning \(2016\)](#) e [Ribeiro, Foss e Cavalheiro \(2017\)](#) trouxeram propostas com base em três conceitos: abstração, análise e automação. Na análise de [Repenning \(2016\)](#), um problema relacionado ao Pensamento Computacional poderia ser dividido em 3 estágios:

- O primeiro estágio refere-se à Formulação do Problema (Abstração): criar uma questão. Como as coisas funcionam? Esta etapa pode ser simplesmente descrita em um pedaço de papel, o que reforça o fato de que o uso do computador é desnecessário para iniciar o processo do Pensamento Computacional.
- No segundo estágio ocorre a Expressão da Solução (Automação): como descrever a solução (através de programação) de forma que um computador possa executá-la.
- E no terceiro estágio, ocorre a Execução e Avaliação (Análise): executando a solução programada no computador, mostrando consequências diretas do próprio pensamento de quem propôs a solução.

Similarmente aos estágios definidos por [Repenning, Ribeiro, Foss e Cavalheiro \(2017, p. 7\)](#) trouxeram as seguintes definições:

Abstração: compreende as abstrações necessárias para dados e processos, e as técnicas de construção de soluções (algoritmos)... Análise: consiste de técnicas de análise de algoritmos quanto a sua correção e eficiência, sob diferentes aspectos ... Automação: envolve a mecanização das soluções (ou de suas partes), permitindo que máquinas nos ajudem a solucionar os problemas.

2.2 Conceitos do Pensamento Computacional

As bases do Pensamento Computacional que foram apresentadas na seção 2.1 trouxeram um conjunto de conceitos que ajudam a compreender e utilizar o Pensamento Computacional. Em adição às teorias apresentadas pelas bases do Pensamento Computacional, [Grover e Pea \(2013\)](#) apresentam outros elementos que são amplamente aceitos como compreendendo uma base curricular que visa apoiar a aprendizagem do Pensamento Computacional e avaliar o seu desenvolvimento:

- Abstrações e generalizações de padrões;
- Processamento Sistemático de Informação;
- Sistemas de símbolos e representações;

- Noções algorítmicas de fluxo de controle;
- Decomposição de problemas estruturados (modularização);
- Pensamento iterativo, recursivo e paralelo;
- Lógica condicional;
- Restrições de eficiência e desempenho;
- Depuração e detecção de erros sistemáticos.

Com relação aos conceitos relativos ao Pensamento Computacional, outra proposta disponível foi a de [Moreno-Leon, Robles e Roman-Gonzalez \(2015\)](#), que trazem sete itens bastante similares aos nove itens apresentados por [Grover e Pea \(2013\)](#):

- Abstração e decomposição de problemas;
- Pensamento/Raciocínio lógico;
- Sincronização;
- Paralelismo;
- Noções algorítmicas de controle de fluxo;
- Interatividade do usuário;
- Representação de dados.

Estes itens apresentados por [Moreno-Leon, Robles e Roman-Gonzalez \(2015\)](#) foram chamados de “Dimensões do Pensamento Computacional” e utilizados como parâmetros para gerar a pontuação do *Dr. Scratch*.

2.3 Atividades Plugadas X Atividades Desplugadas

Com relação às formas de se ensinar o Pensamento Computacional, foi possível encontrar na literatura a referência às atividades plugadas (*plugged*), que são aquelas que ocorrem com a utilização de um computador ou de outro equipamento como um *tablet* ou *Smartphone*, e as atividades desplugadas (*unplugged*), que são aquelas que ocorrem sem o uso de um computador ou similares.

2.3.1 Atividades Plugadas

As atividades plugadas, por utilizarem o computador, podem ser confundidas, em alguns casos, com atividades de programação e, desta forma, deve ser destacado que, quando relacionadas ao Pensamento Computacional, não devem ser compreendidas como uma forma de ensinar programação.

Na realidade, a programação pode ser utilizada de forma a facilitar a compreensão de alguns conceitos do Pensamento Computacional e para aprender Matemática, raciocínio e resolução de problemas, assim como resultados demonstrados por [Clements \(2002\)](#).

Em específico à relação do Pensamento Computacional com a programação, [Csizmadia et al. \(2015\)](#) observam que o raciocínio lógico é fundamental para permitir que alunos avaliem seus códigos e programas.

Utilizar a programação em atividades específicas, como no caso em que os alunos se reúnem em grupos, criam programas e corrigem eventuais erros (*bugs*), podem proporcionar a utilização da abstração e do pensamento algorítmico, por exemplo.

Mas para que a programação possa servir ao propósito de auxiliar as atividades plugadas do Pensamento Computacional, é necessário escolher criteriosamente qual ferramenta, ambiente e/ou linguagem de programação serão utilizadas.

A utilização da programação como ferramenta auxiliar ao ensino do Pensamento Computacional pode ser implementada com o desenvolvimento de jogos, assim como foi proposto por [Kazimoglu et al. \(2012\)](#), que desenvolveram um modelo de jogo no qual os estudantes podiam praticar e desenvolver habilidades do Pensamento Computacional com pouco ou nenhum conhecimento prévio de programação.

Exemplificando um ambiente adequado para utilização em atividades plugadas, [Repenning \(2016\)](#) trouxe a ferramenta *AgentCubes*, que foi utilizada para recriar um jogo clássico do tipo quebra-cabeça, cujos números deveriam ser colocados em ordem.

Programar o jogo proposto pelo autor, utilizando uma linguagem de programação como *Python*, ou *Java*, necessitaria de algumas centenas de linhas de códigos e as linguagens citadas não ofereceriam um ambiente adequado para ensinar conceitos relativos ao Pensamento Computacional.

Já com a utilização do *AgentCubes*, o autor¹² permitiu recriar o jogo de uma forma simplificada, utilizando programação conversacional¹³, que facilitou a tarefa de descrever o segundo estágio¹⁴ do Pensamento Computacional.

¹² Ibidem.

¹³ Forma simples de programação em pares onde o parceiro humano é substituído por um agente computacional chamado Agente de Programação Conversacional (CPA) ([TELLES; HSIEH, 2001](#)).

¹⁴ Que consiste na Expressão da Solução (Automação) conforme descrito na seção 2.1.

As atividades que foram propostas nesta tese¹⁵ para trabalhar com os professores durante a etapa de formação foram do tipo plugadas, utilizando o *Scratch*. A escolha por atividades plugadas foi justificada com base em trabalhos como o de Grover e Pea (2013), que citaram a programação como uma ferramenta chave no apoio às tarefas cognitivas do Pensamento Computacional.

Os autores¹⁶ argumentam que atividades desplugadas fornecem valiosas atividades introdutórias para expor as crianças à natureza da CC. Porém, somente desenvolver atividades desplugadas pode manter os alunos sem contato com experiências computacionais cruciais, envolvidas em práticas comuns do Pensamento Computacional, que ocorrem nas atividades plugadas.

Ainda, a escolha por atividades plugadas incrementa o aprendizado de algumas habilidades de programação que são essenciais no atual contexto tecnológico¹⁷, assim como trouxeram Erni et al. (2016, p. 348):

Com o nível crescente de utilização de computadores em todos os aspectos das vidas humanas, do uso quase invisível nos eletrodomésticos até o uso quase onipresente de *Smartphones* na comunicação, a necessidade de produzir mais programadores nunca foi tão grande.

Outra vantagem na utilização de ambientes como o *Scratch* é que estes, por serem lúdicos, promovem o aprendizado de habilidades do Pensamento Computacional com a vantagem de possuir um nível de complexidade significativamente inferior em comparação às linguagens de programação tradicionais (DODERO; MOTA; RUIZ-RUBE, 2017).

2.3.2 Atividades Desplugadas

As atividades desplugadas podem ser implementadas com materiais simples, como papel e caneta, ou até mesmo com material reciclável, como tampinhas de garrafa. São atividades que não exigem a presença de um laboratório de informática ou equipamentos como computadores e *tablets*.

Exemplos de atividades desplugadas foram propostas no livro¹⁸ de Bell, Witten e Fellows (2011), dentre eles exercícios com números binários, algoritmos básicos e noção de linguagens de programação. Um exemplo sobre algoritmos solicitava que algumas instruções fossem seguidas para desenhar um ponto no centro de uma página:

¹⁵ Descritas no capítulo 4.

¹⁶ Ibidem.

¹⁷ Minha tradução para o texto original: “*With the increasing level of use of computers in all aspects of human lives from the almost invisible use in domestic appliances to the now nearly ubiquitous use of smart phones in communication, the need to produce more programmers has never been greater.*”

¹⁸ A primeira versão proposta deste livro é de 1997.

- Começando da ponta superior esquerda da página, trace uma linha reta passando pelo ponto até a ponta inferior direita;
- Começando da ponta inferior esquerda da página, trace uma linha reta passando pelo ponto até a ponta superior direita;
- Escreva seu nome no triângulo, no centro do lado esquerdo da página.

Seguindo esta sequência de passos (algoritmo) corretamente, o resultado esperado é o que está representado na figura 1.

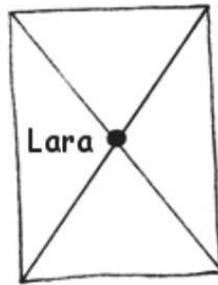


Figura 1 – Saída esperada para o exercício de algoritmos.

Fonte: adaptada de [Bell, Witten e Fellows \(2011\)](#).

Atividades desplugadas são uma ótima alternativa para situações onde não há acesso a uma infraestrutura adequada de informática (laboratórios com computadores disponíveis, acesso à *internet*, entre outros). Permite-se que conceitos do Pensamento Computacional possam ser discutidos e exercitados com materiais simples tais como cartas, tabuleiros e textos.

Ainda, as atividades desplugadas podem ser utilizadas mesmo em ambientes onde há a disponibilidade de infraestrutura adequada de informática, previamente à utilização dos equipamentos (*computadores, tablets, Smartphones*) para promover a introdução de conceitos básicos da programação assim como é o caso dos laços de repetição.

Conforme [Vieira, Passos e Barreto \(2013, p. 672\)](#), a atividade desplugada é uma “[...] técnica que visa ensinar os fundamentos da computação de forma lúdica, sem o uso de computadores, sem distrações e detalhes técnicos em demasia. [...] Um dos objetivos é eliminar as barreiras técnicas e os equívocos sobre o que é realmente a computação”.

A utilização de atividades desplugadas foi extensivamente explorada por [Brackmann \(2017\)](#), trazendo, por exemplo, autômatos desenvolvidos com personagens da Turma da Mônica, e uma versão do jogo Tetris executada com recortes de papel, seguindo instruções para posicionar as peças.

O autor propôs um plano de aula mesclando atividades desplugadas e plugadas, cujas ações consistiam em aulas de *Scratch*.

A utilização de atividades plugadas e desplugadas, de forma mesclada, também foi verificada em outros trabalhos apresentados na literatura. Como exemplo de atividade que utilizou elementos desplugados (cubos feitos de papel e interação física do usuário) e plugados, (*tablets* e *smartphones*) Fuste e Schmandt (2019) utilizam Realidade Aumentada (RA) associada aos conceitos do Pensamento Computacional com uma dinâmica totalmente lúdica.

A atividade permitiu que crianças pudessem experimentar conceitos de programação interagindo com blocos que representavam operações espaciais e físicas.

Além disso, a utilização conjunta de atividades plugadas e desplugadas foi adotada por Berto, Zaina e Sakata (2019), propondo a Metodologia para Ensino de Pensamento Computacional para Crianças (MEPeCoC).

A MEPeCoC trouxe, juntamente com as atividades plugadas e desplugadas, a utilização do desenvolvimento de projetos que funcionou como um mecanismo para que os participantes exercitassem a habilidade de relacionar os conceitos sobre o Pensamento Computacional que foram discutidos durante as atividades.

2.4 Scratch

O *Scratch* foi desenvolvido pelo grupo *Lifelong Kindergarten* do Instituto de Tecnologia de *Massachusetts* (*Massachusetts Institute of Technology* (*MIT*)) e lançado em Maio de 2007 (KINDERGARTEN, 2007) com a proposta de oferecer um ambiente de programação gratuito, no qual é possível criar histórias interativas, jogos e animações.

A utilização do *Scratch* tem se destacado em pesquisas que apoiaram o desenvolvimento do Pensamento Computacional, indicando o mesmo como linguagem potencial a ser utilizada no programa de Ciência da Computação do K-12¹⁹ e de cursos universitários introdutórios à programação (BOE et al., 2013).

O *Scratch* foi criado com base em três princípios de *design* para torná-lo (1) mais adaptável, (2) mais significativo e (3) mais social do que os outros ambientes de programação (RESNICK et al., 2009). Os autores²⁰ trazem que²¹: “À medida que os *Scratchers*²² trabalham em projetos com significado pessoal, descobrimos que eles estão prontos e ansi-

¹⁹ “*K twelve*” é um sistema de ensino público adotado em países como Índia, Estados Unidos da América (EUA) e Canadá, que abrange desde o Jardim de Infância até a 12^a série. Nos EUA (PINTO, 2005) o sistema de ensino obrigatório dura 13 anos desde a *elementary school* (ensino fundamental no Brasil) no qual o aluno ingressa com cinco anos (no *kindergarten*) e fica até a 8^a série e a *High School* (ensino médio no Brasil) da 9^a à 12^a série.

²⁰ Ibidem. p. 65.

²¹ Minha tradução para o texto original: *As Scratchers work on personally meaningful projects, we find they are ready and eager to learn important mathematical and computational concepts related to their projects.*

²² Usuários do *Scratch*.

os para aprender conceitos matemáticos e computacionais importantes relacionados aos seus projetos”. Comprovando que o *Scratch* conseguiu ser significativo para seus usuários.

Em uma pesquisa posterior, também utilizando o *Scratch*, Brennan e Resnick (2012) avaliam o conhecimento de jovens que entraram em contato com a programação dedicando-se a conceitos do Pensamento Computacional como Sequências, *Loops* e Estruturas Condicionais. E mediram a aprendizagem relativa ao Pensamento Computacional utilizando um a técnica chamada²³ “Entrevista baseada em artefatos”.

A técnica consistiu em entrevistar os *Scratchers* com o objetivo de compreender como cada bloco de programação foi utilizado no programa e assim verificar se existiu um real domínio dos conceitos por parte do entrevistado.

A utilização do *Scratch* foi discutida também por Calao et al. (2015), que voltaram-se para diferentes aspectos sobre a aprendizagem de conceitos de programação e o respectivo impacto na aprendizagem matemática.

Os autores supracitados adotam como parâmetros de avaliação o desenvolvimento do raciocínio e a capacidade de formulação de problemas, revelando que houve um significativo aumento na competência dos alunos em resolver problemas matemáticos após utilizarem a programação para solucionar os problemas propostos.

De forma similar, Rocha (2015) apresenta um estudo de caso sobre a construção do conceito de ângulo utilizando programação de labirintos, voltado para a perspectiva do aluno e concluindo sobre a facilidade de utilização do *Scratch* e como o mesmo contribuiu para que os estudantes se apropriassem dos conceitos matemáticos discutidos.

A percepção da facilidade em utilizar o *Scratch* também foi citada por Armoni, Meerbaum-Salant e Ben-Ari (2015), descrevendo o mesmo como um “simples mundo de duas dimensões”, ressaltando, porém, a possibilidade de utilizá-lo para ensinar conceitos avançados da Ciência da Computação (CC) e, conseqüentemente, do Pensamento Computacional.

Para Moraes, Basso e Fagundes (2017), aprender uma linguagem de programação visual como o *Scratch* poderia potencializar as experiências físicas e lógico-matemáticas²⁴, respectivamente, quando estudantes criam os seus projetos e os algoritmos são produzidos.

Já Oliveira (2009) trouxe uma análise da utilização das Tecnologias Digitais da Informação e da Comunicação (TDIC) sob o ponto de vista dos professores ao utilizarem o *Scratch*. Os professores descreveram a facilidade em usar o *Scratch* percebendo-o como um promotor na utilização de *softwares*.

Nesta tese, foi proposta a utilização do *Scratch* como ferramenta na formação de professores, uma vez que a utilização do *Scratch* permite trabalhar todas as dimensões

²³ Minha tradução para o texto original “*Artifact-Based Interviews*”.

²⁴ Chamadas assim por Piaget.

do Pensamento Computacional, conforme destacado por Moreno-Leon, Robles e Roman-Gonzalez (2017). A avaliação dos programas desenvolvidos no *Scratch* pode ser efetuada empregando o *Dr. Scratch*.

O *Scratch* pode ser utilizado em dois modos: (1) no modo *online*²⁵, apresentando vantagens como permitir ao usuário compartilhar na nuvem o projeto criado ou acessar projetos de outros usuários para aprender; e (2) no modo *offline*, instalando o *Scratch* localmente no computador.

Na figura 2, apresenta-se o Ambiente de Desenvolvimento Integrado²⁶ do *Scratch* (modo *online*).

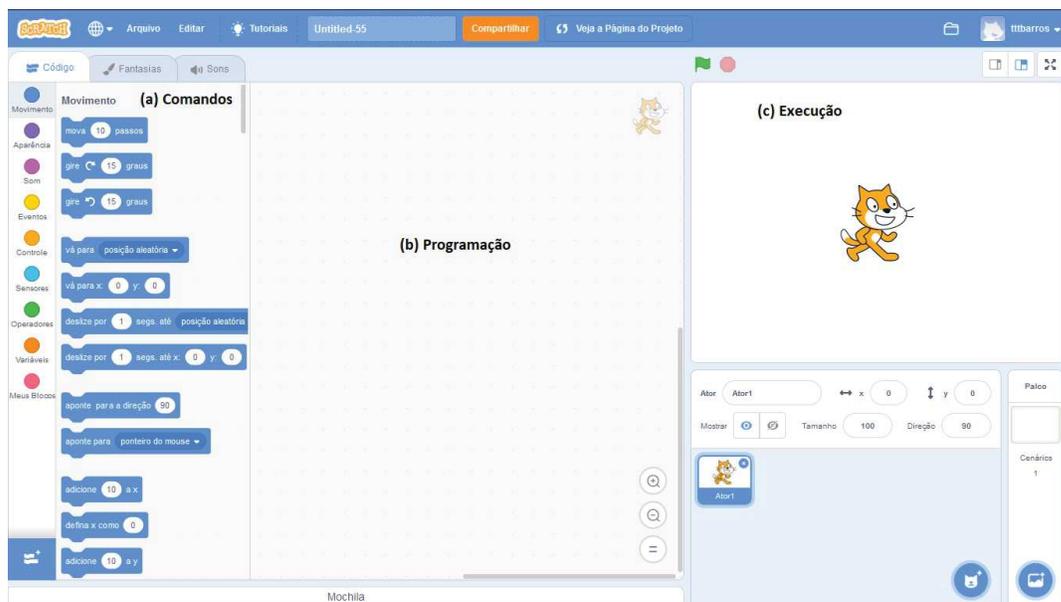


Figura 2 – IDE do *Scratch*.

Fonte: elaborada pelo autor.

Podem-se observar três áreas principais da IDE, conforme apresentado na figura 2:

- (a) Área dos Comandos: onde são disponibilizados os blocos de programação. Cada bloco representa um comando ou instrução de programação, como, por exemplo, o comando “mova”, que desloca o ator na tela de execução por determinado número de passos. Pode-se observar que estes blocos estão divididos em categorias com cores diferenciadas, o que ajuda a compreender como mesclar blocos de diferentes categorias para obter resultados;
- (b) Área de Programação: onde os diferentes blocos serão conectados para criar o programa. É onde o programa é efetivamente criado;

²⁵ Acessando <https://scratch.mit.edu/>.

²⁶ *Integrated Development Environment* - IDE.

- (c) Área de Execução: é onde o resultado do programa pode ser visualizado, ou onde o programa “roda” - conforme o jargão de programação.

Ainda, outros elementos presentes na figura 2 são fundamentais para a compreensão de como utilizar o *Scratch* como, por exemplo, o Ator representado pelo gato; o plano de fundo ou palco, que, neste caso, está em branco mas poderia ser configurado tal como uma imagem do fundo do mar para criar uma animação sobre animais marinhos.

O próprio *Scratch* pode ser considerado como um exemplo de abstração, assim como a analogia proposta por [Armoni e Ben-Ari \(2013\)](#), pois quando utilizamos um comando como o “move”, não há necessidade em compreender o funcionamento do computador e como o movimento do ator na tela está sendo calculado. Simplesmente haverá visualização do ator se deslocando nos eixos horizontal e/ou vertical que possuem 480 e 240 unidades, respectivamente.

Os detalhes sobre resolução da tela, ou como o comando *move* é interpretado não são importantes, ou seja, podem ser abstraídos e o *Scratcher* (programador) precisa somente conhecer o comando e o resultado esperado do mesmo.

2.5 *Dr. Scratch*

O *Dr. Scratch* é um aplicativo *Web* gratuito e de código aberto que permite analisar projetos criados no *Scratch*, gerando informações sobre o grau de desenvolvimento de cada um dos conceitos do Pensamento Computacional.

Uma pontuação é gerada a partir da avaliação da competência demonstrada pelo desenvolvedor do projeto com relação aos conceitos de: abstração e decomposição de problemas; raciocínio lógico; sincronização; paralelismo; controle de fluxo; interatividade do usuário; e representação de dados.

A Tabela 1 mostra as regras utilizadas para gerar a pontuação referente aos conceitos do Pensamento Computacional. A pontuação pode variar de 0 a 21 pontos.

Os resultados gerados pelo aplicativo consideram o nível do Pensamento Computacional detectado no programa avaliado. Estes resultados ajudam *Scratchers* iniciantes com informações básicas das melhorias mais importantes a serem executadas no programa.

À medida que a pontuação dos programas avaliados aumentam, mais informações sobre os projetos analisados são geradas e, assim, usuários avançados recebem um relatório de *feedback* com informações tanto em termos de habilidades do Pensamento Computacional, como em relação aos maus hábitos de programação.

A figura 3 mostra uma tela gerada pelo *Dr. Scratch* referente à avaliação de um programa que pontuou seis pontos dos vinte e um possíveis, sendo classificado como de

Tabela 1 – Dimensões do Pensamento Computacional.

Dimensões do Pensamento Computacional	Nível Básico (1 ponto)	Nível em Desenvolvimento (2 pontos)	Nível Proficiente (3 pontos)
Raciocínio Lógico	Se	Se Senão	Operações Lógicas
Representação de Dados	Modificadores de propriedades dos objetos	variáveis	listas
Interatividade do Usuário	Bandeira verde	Teclado, <i>mouse</i> , Perguntar e Espere	<i>WEBCAM</i> , <i>Microfone</i>
Controle de Fluxo	Sequência de Blocos	Repita n vezes, Sempre	Repita até que
Abstração e Decomposição de Problemas	mais de um <i>SCRIPT</i> e mais de um ator	Procedimentos (definição de blocos)	uso de clones
Paralelismo	2 <i>Scripts</i> na bandeira verde	2 <i>Scripts</i> para determinada tecla ou ator clicados	2 <i>Scripts</i> no recebimento de mensagens, ou vídeo ou microfone, ou na troca de plano de fundo
Sincronização	Espere	Mensagem, Pare tudo, Parar programa	Espere até, Quando o plano de fundo mudar para, Mensagem e Espera

Fonte: adaptado de (MORENO-LEON; ROBLES; ROMAN-GONZALEZ, 2015; MORENO-LEON; ROBLES; ROMAN-GONZALEZ, 2017) com minha tradução.

nível básico.

Além disso, há a informação que, dos seis pontos obtidos, dois pontos foram relativos à interatividade do usuário (dois de três pontos possíveis), dois pontos foram relativos à representação de dados (dois de três pontos possíveis), e mais dois pontos foram relativos ao controle de fluxo (dois de três pontos possíveis).

Com estas informações, foi possível avaliar o grau de desenvolvimento do programa avaliado e ter a noção de qual item do Pensamento Computacional foi mais desenvolvido. As informações geradas no *Dr. Scratch* foram utilizadas nas seções 4.1.7 e 4.4 para classificar, respectivamente, os programas criados pelos professores na etapa do curso de formação e dos programas criados pelos professores e alunos na etapa de visitaç o nas escolas, avaliando, por exemplo, qual conceito do Pensamento Computacional obteve maior pontuaç o e, conseq entemente, teve maior desenvolvimento nos grupos.

As an lises realizadas nas seções 4.1.7 e 4.4 foram similares   an lise realizada por Šerbec, Cerar e Žerovnik (2018), que buscaram compreender a diferenç a no racioc nio de alunos do ensino fundamental com relaç o aos alunos graduandos de Ci ncia da Computa o durante o desenvolvimento de programas no *Scratch*.

Os autores concluíram que a utilizaç o do *Dr. Scratch* demonstrou um resultado j  esperado, no qual os graduandos obtiveram maiores pontuaç es e que, se a programaç o reflete o modo de pensar, ent o o Pensamento Computacional poderia ser avaliado pela

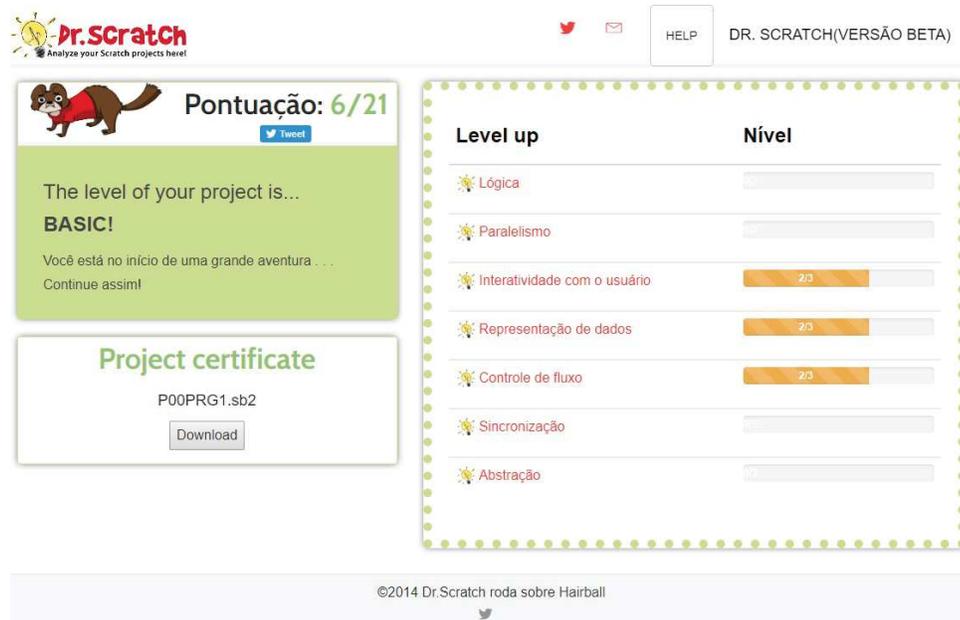


Figura 3 – Relatório gerado pelo *Dr. Scratch*.

Fonte: elaborada pelo autor.

qualidade dos programas que o aluno produziu.

Considerações sobre o Capítulo

O Pensamento Computacional é um conceito amplo e possui uma série de definições propostas por diferentes autores. Ao escrever o texto desta tese tive a oportunidade de avaliar as conceituações propostas na literatura e assim formular o seguinte conceito para o Pensamento Computacional: “Conjunto de habilidades que permitem ao ser humano propor a solução de um problema de forma sistematizada e replicável por outros seres humanos ou por computadores”.

As bases do Pensamento Computacional permitem avaliar a abrangência que sua utilização possui nas áreas afins, como computação e Informática, e similarmente em áreas como Ciências Humanas, Ciências Exatas e Artes.

É fundamental destacar que o ponto a ser discutido não é que todos precisem conhecer o Pensamento Computacional para poder se tornar um cientista da computação, programador, ou mesmo atuar na área da tecnologia, mas que todos deveriam conhecer o Pensamento Computacional e seus conceitos, pois poderiam beneficiar-se destes conhecimentos, independentemente de sua área de atuação.

A mesma abstração, que é uma das bases do Pensamento Computacional, é a abstração que permite a um artista criar uma escultura ou aplicar um tom de cor em uma aquarela, embora em contextos diferentes.

Da mesma forma, o reconhecimento de padrões pode contribuir para que um programador reutilize uma função já escrita, ou que um médico identifique um agrupamento celular identificando o mesmo como um tumor.

E assim como um leigo em Medicina pode aprender sobre os sistemas digestivos e respiratórios sem necessariamente ter que se formar na área da saúde, o sistema educacional pode oferecer conhecimentos básicos de Pensamento Computacional sem necessariamente prever que todos tenham que seguir uma carreira na área tecnológica como CC ou Engenharia.

O Pensamento Computacional pode ser introduzido já nas séries iniciais do Ensino Fundamental, quando as atividades desplugadas podem ser uma boa abordagem, além de oferecerem uma alternativa para os casos em que não haja acesso ao computador.

Já as atividades plugadas, assim como as atividades que foram propostas nesta tese, exigem uma infraestrutura adequada como um laboratório e computadores disponíveis, permitindo o desenvolvimento de atividades com *Softwares* específicos e o contato com conceitos fundamentais, inclusive da CC.

3 Formação de Professores

Introdução

A discussão sobre o sistema educacional tanto no Brasil, como no mundo é ampla e possui uma vasta literatura na qual se apresenta, por exemplo, a preocupação com a forma como ensinamos nossos alunos, qual a estrutura mais apropriada da grade curricular e quais as atividades mais apropriadas para cada faixa etária em relação a cada saber, dentre outros tópicos.

Ainda, um tópico amplamente discutido dentro do contexto educacional se refere aos docentes e sua formação, dada a importância fundamental que a atuação de um professor possui na vida de seus alunos.

O papel de ensinar é extremamente importante, pois, independentemente da fase escolar em que uma pessoa esteja, o professor deve ser aquele que oferece um caminho no qual seu aluno pode seguir. Lembro-me da minha primeira professora do Ensino Fundamental, que, no meu primeiro dia de aula, pegou minha mão e me conduziu até a sala de aula, mostrando-me, posteriormente, o “a” de abelha. O aprendizado do alfabeto foi o primeiro passo que, somado aos demais conhecimentos adquiridos ao longo de muitos anos, me permitiu redigir o texto desta tese.

Também guardo recordações de meus professores engenheiros, reconhecidos como expoentes em suas áreas de atuação dado o extenso domínio sobre conteúdos como cálculo, programação e robótica.

Tenho convicção que, além do amor pela profissão, todos os meus professores possuíam uma característica em comum: um profundo conhecimento dos assuntos que ensinavam. Para ensinar, um professor não pode simplesmente se aventurar sem estar preparado. Mas precisa, sim, estudar e capacitar-se constantemente, assim como orientou Freire (2001, p. 259) em sua carta aos professores:

O fato, porém, de que ensinar ensina o ensinante a ensinar um certo conteúdo não deve significar, de modo algum, que o ensinante se aventure a ensinar sem competência para fazê-lo. Não o autoriza a ensinar o que não sabe. A responsabilidade ética, política e profissional do ensinante lhe coloca o dever de se preparar, de se capacitar, de se formar antes mesmo de iniciar sua atividade docente. Esta atividade exige que sua preparação, sua capacitação, sua formação se tornem processos permanentes. Sua experiência docente, se bem percebida e bem vivida, vai deixando claro que ela requer uma formação permanente do ensinante. Formação que se funda na análise crítica de sua prática. Partamos da experiência de aprender, de conhecer, por parte de quem se prepara para a tarefa docente, que envolve necessariamente estudar.

Sobre a formação de professores no Brasil, apresenta-se no material divulgado pelo Instituto Ayrton Senna (2014, p. 18) um primórdio das ações referentes à formação de professores, publicada em 1827, quando foi apresentado o “Primeiro curso para preparar professores para as escolas de primeiras letras, com estudos voltados aos conteúdos a serem ensinados na escola de primeiras letras e recomendações relativas às metodologias de ensino”.

Além disso, essa obra ¹ trouxe um alinhamento com a ideia de Freire (2001) sobre a necessidade do professor estar constantemente aprendendo e se capacitando:

[...] professor, assim como qualquer profissional, precisa de constante atualização para obter bons resultados no trabalho. A profissão é dinâmica e exige que o docente seja um eterno aprendiz. Portanto, a formação continuada deve ser parte integrante de sua vida profissional.

Adicionalmente, para trazer uma dimensão temporal sobre a questão da formação dos professores, o trabalho de Mello (2000), que remonta ao Século XIX, parece atual quando comparado a publicações mais recentes². A autora discutiu sobre o sistema brasileiro de formação de professores e suas inadequações para fazer cumprir a Lei de Diretrizes e Bases da Educação Nacional (LDB).

Trazendo também³ uma excelente reflexão sobre o aprender a aprender e sua relação com a formação dos professores:

No futuro, a boa qualidade dos professores poderá eliminar os custos de organização dos grandes empreendimentos de capacitação ou educação continuada destinados a ensinar àqueles que, se tivessem aprendido a aprender, poderiam ser gestores da própria atualização profissional.

A reflexão de Mello (2000) colocou o professor no papel principal de gestor (e, conseqüentemente, responsável) de sua própria formação. A autora também apresentou⁴ o perfil do docente reflexivo⁵:

Ensinar, portanto, exige aprender a inquietar-se e a indignar-se com o fracasso sem deixar destruir-se por ele. Essas competências traçam o perfil do profissional denominado reflexivo [...]: um profissional cuja atuação é inteligente e flexível, situada e reativa, produto de uma mistura integrada de ciência, técnica e arte, [...]. O profissional reflexivo é também aquele que sabe como suas competências são constituídas, é capaz de entender a própria ação e explicar por que tomou determinada decisão, mobilizando para isso os conhecimentos de sua especialidade.

¹ Ibidem. p. 34.

² Assim como Coimbra (2020) que citou a LDB e analisou as resoluções do Conselho Nacional da Educação (CNE) elucidando por exemplo para qual sociedade retrata-se a formação de professores.

³ Ibidem. p. 101.

⁴ Ibidem. p. 104.

⁵ Tema apresentado também em trabalhos como (ROMAN, 2013; FAGUNDES, 2016) alinhados à formação de professores.

Com a ideia estabelecida sobre a importância da formação de professores, o trabalho de [Gatti \(2010\)](#) apresentou características e problemas da formação de professores discorrendo sobre os currículos dos cursos de licenciaturas e as características dos licenciandos. A autora constatou que entidades profissionais e científicas opõem resistência às soluções de caráter interdisciplinar para o currículo frente à forte tradição disciplinar que caracteriza a identidade docente e que seria necessária uma revolução nas estruturas institucionais formativas e nos currículos de formação.

Adicionalmente, o documento “Formação de Professores no Brasil - Diagnóstico, agenda de políticas e estratégias para a mudança⁶”, coordenado por [Abrucio \(2016\)](#), trouxe um rigoroso estudo sobre formação de professores.

O objetivo deste documento foi encontrar diagnósticos e medidas consensuais articulando-os na construção de propostas e estratégias de aperfeiçoamento do processo formativo dos professores desde a fase inicial, até a fase da educação continuada.

Dentre os resultados apresentados no trabalho⁷, foram citados sete problemas principais com relação à formação docente: 1. integração do tripé formativo (universidades-centros formadores/redes de ensino/escolas); 2. perfil do aluno que, futuramente, poderá ser professor ; 3. currículo da Educação Básica; 4. currículo e estrutura profissional-pedagógica dos cursos de pedagogia e licenciaturas; 5. Educação a Distância; 6. profissionalização da prática docente, da formação inicial à continuada; e 7. atratividade/motivação da carreira docente.

Na discussão sobre os problemas apresentados, [Abrucio \(2016\)](#) trouxe, com relação ao item⁸ 4, que ao contrário do que acontece em outros países onde o aluno é protagonista e avaliado por competências em um ambiente interdisciplinar e de metodologias ativas, no Brasil não ocorre esta discussão dentro das universidades.

Por isso, ações sobre interdisciplinaridade e metodologias ativas aparecem pouco nas escolas públicas brasileiras. Além do mais, o autor conclui a discussão reforçando a argumentação de [Gatti \(2010\)](#) sobre a necessidade da revolução nas estruturas institucionais formativas e nos currículos da formação.

Ou seja, analisando-se as colocações de [Gatti \(2010\)](#) e [Abrucio \(2016\)](#), foi possível compreender que a formação de professores no Brasil ainda precisa de muito espaço de discussão e de ações - principalmente governamentais - para gerar resultados eficientes em médio e longo prazos.

Sendo esta tese um trabalho com teor fortemente tecnológico, buscou-se na litera-

⁶ Iniciativa do movimento Todos Pela Educação (<https://www.todospelaeducacao.org.br/>) cuja missão é contribuir para que até 2030, o país assegure educação Básica pública de qualidade a todas as crianças e jovens. Sendo uma de suas cinco bandeiras a melhoria da formação e carreira do professor.

⁷ Ibidem. p. 33-34.

⁸ 4. currículo e estrutura profissional-pedagógica dos cursos de pedagogia e licenciaturas.

tura discussões sobre a relação da formação dos professores frente às ferramentas tecnológicas. Trazendo como exemplo o trabalho de Moran (2000) que destacou a importância da formação do professor frente à tecnologia.

O autor contextualizou a questão de que passamos muito rapidamente do livro para a televisão e vídeo, e destes para o computador e *internet* sem aprender a explorar todas as possibilidades de cada meio. Dentro deste contexto, onde o papel de explorar estas tecnologias deveria estar obrigatoriamente associado à formação do professor, o qual deveria compreender a importância do seu papel na utilização destas tecnologias e se apropriar destas. Mesmo que isto implicasse em uma mudança em suas práticas pedagógicas em concordância com Prado (2005, p. 13):

Uma realidade com a qual o professor depara atualmente é caracterizada pela chegada de novas tecnologias (computador, Internet, vídeo, televisão) na escola, que apontam novos desafios para a comunidade escolar. O que fazer diante desse novo cenário? De repente, o professor que, confortavelmente, desenvolvia sua ação pedagógica – tal como havia sido preparado durante sua vida acadêmica e pela sua experiência em sala de aula – se vê diante de uma situação que implica novas aprendizagens e mudanças na prática pedagógica.

Foi possível observar que tanto Moran (2000) como Prado (2005) trouxeram uma mesma realidade quanto à questão da necessidade de o professor ser um sujeito ativo, que precisa se apropriar das tecnologias para que estas possam ser então utilizadas na sua prática pedagógica.

Inclusive, se avaliarmos os trabalhos cronologicamente, uma mesma realidade sobre a formação dos professores, frente à tecnologia, foi apresentada em 2000 e 2005. À medida que os anos se passaram, a questão da necessidade de formar professores compatíveis com o cenário tecnológico parece não ter evoluído, ou, sendo otimista, parece ter evoluído pouco.

Conforme apresentado na Revista Educação (2018) - que foi publicada 13 anos após o trabalho de Prado (2005) e 18 após o trabalho de Moran (2000)) - foi possível deparar-se com um cenário análogo sobre a formação dos professores em relação à tecnologia:

O maior desafio atualmente é os professores conseguirem notar que a tecnologia pode tornar o processo de ensino-aprendizagem melhor [...] parte da desconfiança de alguns docentes com relação ao uso das novas tecnologias vem das mudanças que elas causam na própria rotina da aula. “É algo que tira o professor da zona de conforto. É uma ferramenta que precisa de estudo em casa, de um planejamento maior, de um período semanal que exige reflexão e estudo.” Outro fator que gera desconfiança é o medo de a tecnologia atuar como um distrator. No uso da internet, por exemplo, o receio é que os alunos acabem desviando a atenção do conteúdo para as redes sociais. [...] driblar o problema também passa pela formação docente. “O professor precisa dominar essas ferramentas, participar de cursos, se inteirar a respeito, praticar. É preciso estar embasado para manter a atenção do aluno” [...].

Tomando apoio na citação cronológica aos trabalhos relacionados, buscou-se uma referência anterior ao ano 2000 para verificar como a relação da formação de professores diante da tecnologia se apresentava no início da transição da era da *Internet* no Brasil. E o trabalho de Cysneiros (1996) trouxe uma boa ideia de como estava esta realidade:

Na primeira versão deste trabalho (1996), meu ponto de partida era uma objeção comum, que ouvi muitas vezes de professores que vivenciam diariamente os problemas da escola pública brasileira: como colocar computadores em uma escola que não possui condições materiais básicas, não tem pessoal de apoio nem professores capacitados e com tempo para usá-los? [...] O professor típico nem sempre domina o conteúdo que ensina, encontra-se sobrecarregado com aulas em mais de uma escola, falta-lhe tempo para estudar e experimentar coisas novas [...]

Neste caso, o autor⁹ publicava um trabalho na década de 90, quando o acesso ao computador pessoal ainda estava “engatinhando”, principalmente nas escolas públicas, e então, considerou-se compreensível que a preparação dos professores para absorver os conhecimentos tecnológicos fosse defasada, principalmente aqueles que estavam habituados ao uso do computador.

Porém, o que se considerou preocupante foi escrever sobre assunto similar no ano de 2020 e, tomando por base trabalhos científicos, perceber que houve pouca evolução ao longo dos anos.

Da mesma forma como Cysneiros (1996) relatava o problema da formação tecnológica de professores, o autor também mostrava uma preocupação com os alunos que deveriam estar inseridos no contexto tecnológico: “Os alunos de escolas públicas estarão muito em breve no mercado de trabalho (muitos já estão) e serão os cidadãos das primeiras décadas do próximo século, onde o uso das TI será ainda mais corriqueiro [...]”.

O uso corriqueiro da TI chegou e nós, brasileiros, ao que tudo indica, não conseguimos preparar nossos alunos para assumirem a demanda tecnológica atual, se considerarmos a escassez de profissionais de TI em um mercado que estima¹⁰ 70 mil vagas até 2024, com a possibilidade de que estas vagas possam não ser preenchidas devido à falta de profissionais capacitados.

Sendo assim, podemos pensar na relação entre a carência na formação dos professores e no impacto causado no desempenho dos alunos no contexto tecnológico atual.

Logo, os professores, possuindo carência em sua formação, não conseguiram disseminar os conhecimentos tecnológicos básicos aos seus alunos. Estes alunos, consequentemente, acabaram por não seguir uma carreira tecnológica, demonstrando o impacto e importância da formação dos professores.

⁹ Ibidem.

¹⁰ Informação extraída de <https://bit.ly/vagasTecnologia>, acessado em 15/12/2019.

Talvez uma explicação para o distanciamento do professor, em relação à tecnologia, esteja nas palavras de Perrenoud (1999, p. 6), propondo que a evolução do trabalho docente transcorre em um ritmo menor que o das mudanças sociais e tecnológicas:

[...] a despeito das novas tecnologias, da modernização dos currículos, da renovação das ideias pedagógicas, o trabalho dos professores evolui lentamente porque depende pouco do progresso técnico, porque a relação educativa obedece a uma trama bastante estável e porque suas condições de trabalho e sua cultura profissional instalam os professores em rotinas. [...] Um viajante que voltasse à vida depois de um século de hibernação veria a cidade, a indústria, os transportes, a alimentação, [...] consideravelmente mudadas. Entrando numa escola, ao acaso, encontraria uma sala de aula, um quadro-negro e um professor dirigindo-se a um grupo de alunos. [...] não haveria nenhuma dúvida de que encontrava-se em uma escola. Talvez houvesse um computador na sala, conectado a uma rede. Mas o visitante observaria que ele é usado para propor exercícios na tela e preparar conferências “surfando” em páginas da Web. O triângulo didático estaria no lugar, [...]

Mas, obviamente, não se pode colocar toda a responsabilidade pelo processo de formação docente no próprio docente. A escola, diretores, coordenação pedagógica, demais estruturas de apoio e, principalmente, o Estado devem garantir condições para o docente aproveitar as formações para que, em seguida, tenha um espaço para testar o que aprendeu e consiga implantar as mudanças em sala de aula.

A carreira de professor infelizmente passa por diversas carências, tomando como exemplo o relato de Jacomini e Penna (2016, p. 185): “Ao mesmo tempo em que se exige muito dos professores, se oferecem poucas recompensas monetárias e pouco reconhecimento social”.

O autor destacou que fatores como o acúmulo de cargos, prática comum na carreira docente, acaba por impactar na saúde do professor e, conseqüentemente, na qualidade da educação, já que um professor sobrecarregado não possui condições de preparar uma boa aula¹¹:

[...] em acúmulo de cargo, o professor poder trabalhar até 70 horas, o que implica numa jornada diária de 14 horas, cinco dias por semana. Duas questões podem ser destacadas: maior possibilidade de adoecimento do professor e um exercício docente aquém do potencial do profissional, como consequência do excesso de trabalho. Ambas têm implicação na qualidade do ensino, seja pelo excesso de licença médica, seja por aulas que nem sempre são bem preparadas ou bem ministradas, em consequência do cansaço do professor. Disso decorre que, entre as muitas questões que interferem na qualidade da educação escolar, está a jornada de trabalho do professor.

¹¹ Ibidem. p. 189.

Destaca-se que a proposta que foi apresentada nesta tese envolveu tanto a realização de um curso de formação em Pensamento Computacional para os professores, como a posterior verificação da forma como estes utilizariam a formação em sala de aula.

Sendo assim, pretendeu-se verificar¹² espaço, infraestrutura e apoio dentro das escolas dentre outros, que permitam ao docente aplicar o que for assimilado na formação.

Além das questões específicas relacionadas ao docente, diversos trabalhos ((SILVA, 2015; SÁ; WERLE, 2017; ECHALAR; PEIXOTO, 2017; SILVA; SILVA; FRANÇA, 2017)) trataram dos problemas relacionados à infraestrutura deficiente das escolas oferecida aos professores, fato que, em muitos casos, culminou em um desempenho abaixo do esperado do professor que participou de determinada formação. Pois desprovido de uma infraestrutura adequada, o professor não teve a oportunidade de colocar em prática o que aprendeu em sala de aula porque não disponibilizava dos equipamentos e materiais necessários e/ou porque não disponibilizava de tempo para preparar seu material e trabalhar com seus alunos.

Este tipo de cenário, onde o professor não dispõe de recursos, não é uma realidade exclusiva do Brasil, assim como pode ser encontrado no trabalho de Bingimlas (2009) que citou a falta de suporte técnico nas escolas como uma das dificuldades para a integração das TICs junto aos professores.

Com o termo suporte técnico referindo-se aos recursos disponíveis para o professor na escola. E como a publicação se tratava de uma revisão da literatura, esta trouxe a realidade de vários países, como da Austrália, por exemplo.

Outro tópico concernente à formação de professores, no contexto tecnológico, foi a dificuldade apresentada pelos professores na inserção das TICs em sala de aula, sendo esta dificuldade uma forte barreira, assim como apresentado por Schuhmacher, Filho e Schuhmacher (2017, p. 575), que aconselharam sobre a forma de conceber um currículo que tratasse da utilização das TICs de forma que sejam aproveitadas pelos docentes de uma maneira mais técnica e não somente como um apoio à organização de sua sala de aula:

[...] é necessário que, nos cursos de formação inicial e continuada, a preocupação se estenda para além de currículos que apresentem conteúdos nos quais o foco principal é o uso de ferramentas que organizem a prática docente, ou seja, ferramentas de produção e domínio dos recursos primários do computador. Entender o professor formador como um sujeito que possui conhecimentos construídos sobre as TIC e apresenta opiniões sobre a mesma, deve ser considerado como fundamental para a superação e ruptura de conhecimentos incompletos e/ou falsos.

A questão da formação de professores, mais especificamente este saber relacionado

¹² Assim como foi proposto por Jacomini e Penna (2016).

à tecnologia, ainda é um campo fértil para a pesquisa e a formulação de soluções. Mas que, infelizmente, apresenta pequenos resultados em relação aos grandes esforços relatados nos trabalhos científicos concernentes.

Assim, conforme o Objetivo Geral e objetivos específicos desta tese, a mesma foi alinhada com a problemática da formação dos professores no contexto tecnológico. Mais especificamente, buscando compreender como os professores de Matemática e Informática, dos anos finais do Ensino Básico, poderiam se apropriar dos conhecimentos do curso de formação em Pensamento Computacional e como estes docentes poderiam aplicar os conhecimentos adquiridos em sala de aula.

3.1 Formação de Professores em Relação ao Pensamento Computacional

Uma vez consolidada a ideia sobre a importância do Pensamento Computacional e da necessidade de termos uma boa formação de nossos professores, pode-se voltar a atenção para verificar como ocorre a formação de docentes em relação ao Pensamento Computacional.

Assim, pode-se referenciar o trabalho de [Hu \(2011\)](#), o qual avaliou que pensar computacionalmente é tão fundamental quanto ler, escrever e realizar operações matemáticas básicas. Desta forma, o sistema educacional poderia estar em xeque caso os professores não consigam chegar em um consenso sobre o conceito de Pensamento Computacional e como ensinar este conceito.

No trabalho de [Yadav et al. \(2014\)](#) foi apresentada uma pesquisa sobre os professores do K-12¹³ e como deveria ser sua preparação a fim de apresentar os conceitos do Pensamento Computacional para os alunos.

Desta maneira, os alunos podem passar de simples “consumidores” de tecnologia para indivíduos capacitados em produzir tecnologia, cuja consciência permitirá aplicar suas próprias ideias. Porém, os autores destacaram que iniciativas apresentadas para educar sobre o Pensamento Computacional ainda eram limitadas aos professores da área da CC.

Esta tese concentra-se em professores de Matemática e Informática como sujeitos de pesquisa, justificando-se esta escolha devido à carência de conhecimentos prévios na área computacional da grande maioria das demais formações.

Assim como foi discutido por [Curzon et al. \(2014, p. 89\)](#), com referência a trabalhos que trouxeram iniciativas em relação ao Pensamento Computacional¹⁴: “Um grande

¹³ Cf. seção 2.4.

¹⁴ Minha tradução para o texto original: “A big issue facing these initiatives is that not all the teachers

problema enfrentado por essas iniciativas é que nem todos os professores envolvidos têm formação em computação”. O autor considerou, uma vez que o Pensamento Computacional foi o assunto central das iniciativas, que os professores precisavam de um profundo conhecimento para permitir que fossem traçadas formas de auxiliar seus alunos.

A proposta de [Curzon et al. \(2014\)](#) consistiu na realização de quatro *Workshops*, onde, dentre outros assuntos, se discutiu sobre os conceitos de decomposição, abstração e pensamento algorítmico utilizando atividades desplugadas como truques com cartas.

Nas conclusões do trabalho, o autor citou que não houve uma avaliação referente à utilização dos conhecimentos adquiridos pelos professores, cujo caso ideal seria avaliar posteriormente aos *Workshops* de que forma os professores utilizassem o que aprenderam durante as atividades.

Destaca-se que, nesta tese, assim como apresentado na seção 4.2, houve uma etapa de visitação às escolas, posterior à etapa de formação dos professores, permitindo avaliar como estes utilizaram os conhecimentos adquiridos em sala de aula com seus alunos.

Similarmente aos conceitos apresentados por [Curzon¹⁵](#), em [Csizmadia et al. \(2015\)¹⁶](#), os autores criaram um guia sobre o Pensamento Computacional para os professores afeitos aos conceitos de Pensamento Algorítmico, Decomposição, Reconhecimento de Padrões e Abstração. Ainda, foram apresentadas as técnicas de Reflexão, Codificação, Projeto, Análise e Aplicação associadas com o Pensamento Computacional.

O documento trouxe uma contextualização sobre como utilizar os conceitos e técnicas apresentados em sala de aula, identificando comportamentos distintos dos estudantes que permitiam identificar a assimilação de cada conceito.

Sendo o Pensamento Computacional uma competência fundamental do Século XXI, [Mandaji et al. \(2018\)](#) trazem que a atual formação (acadêmica) dos professores não contempla itens como aprendizagem criativa, resolução de problemas e capacidades cognitivas para vencer o desafio de implantar o Pensamento Computacional na escola¹⁷.

Os autores supracitados destacam o papel do professor como sujeito principal na elaboração e implementação de um currículo, tendo a função de contextualizar e dar sentido aos aprendizados por meio dos seus conhecimentos e práticas e por sua relação estabelecida com os estudantes. Desta forma, a literatura pesquisada demonstrou que há uma preocupação com as questões relacionadas ao Pensamento Computacional e como este tem sido trabalhado junto aos professores.

Algumas iniciativas demonstram que a formação em Pensamento Computacional,

involved have computing backgrounds.”.

¹⁵ Ibidem.

¹⁶ Do qual Paul Curzon também é autor.

¹⁷ Sendo esta conclusão (em termos gerais) muito similar ao que concluíram [Gatti \(2010\)](#) e [Abrucio \(2016\)](#) na seção de Introdução do capítulo 3.

para os professores, já foi um assunto discutido e trabalhado, assim como apresentado pela organização *Digital Promise*¹⁸ que trouxe em um de seus documentos as seguintes recomendações para promover a formação de professores em ambientes educacionais onde existisse a intenção de implantar atividades referentes ao Pensamento Computacional (Digital Promise, 2017, p.27):

- Expandir caminhos inovadores para o aprendizado profissional, incluindo micro-crédito adicional, para apoiar educadores de diferentes faixas etárias e áreas de conteúdo;
- Desenvolver e dimensionar oportunidades para desenvolvimento profissional de professores em serviço, tanto *online* como presencialmente;
- Criar recursos para apoiar administradores escolares que estão integrando o Pensamento Computacional em suas escolas;
- Desenvolver estratégias prontas para implementação de professores e líderes escolares para tornar o Pensamento Computacional mais inclusivo, equitativo e útil para a diversidade completa de alunos (culturalmente, neurologicamente, etc.).

3.1.1 Trabalhos relacionados

Com relação aos trabalhos relacionados especificamente à formação dos professores no contexto do Pensamento Computacional, Silva, Silva e França (2017) descrevem uma formação em Pensamento Computacional com atividades desplugadas (como a “carta mágica”¹⁹) e plugadas utilizando *Scratch* e o *Lightbot*²⁰, cuja formação foi oferecida para 13 participantes com formação pedagógica e tecnológica que lecionavam cursos de Robótica e de *Internet*. As conclusões do trabalho apontam a desmotivação dos professores por desconhecimento da temática do Pensamento Computacional, fato este sanado durante o curso.

Após os professores tomarem conhecimento sobre o Pensamento Computacional, os mesmos ficaram motivados para multiplicar os conhecimentos adquiridos. Os autores trazem a disseminação do Pensamento Computacional na formação continuada de professores da educação básica como sugestão de trabalhos futuros. Investiga-se como estes professores pretendiam integrar o Pensamento Computacional em sua prática pedagógica.

¹⁸ Criada com a missão de acelerar a inovação na educação, após constatação das lacunas educacionais que surgem quando alunos não possuem acesso igualitário a uma nova tecnologia ou recurso de aprendizagem. E como a educação americana possui exemplos de excelência mundial, a *Digital Promise* busca expandir as mesmas condições à toda a sociedade <https://digitalpromise.org/>.

¹⁹ Esta atividade pode ser verificada em https://bit.ly/carta_magica.

²⁰ É um aplicativo que “ensina programação secretamente enquanto você joga”, disponibilizado em <https://lightbot.com/>.

Destaca-se que nesta tese a proposta apresentada foi consonante com a proposta de Silva, Silva e França (2017) no sentido de formar professores da educação básica, verificando após a etapa de formação como os professores buscaram a integração em sala de aula.

Com o diferencial que as atividades utilizadas pelos autores²¹ foram mistas entre plugadas e desplugadas, enquanto nesta tese optou-se por utilizar exclusivamente atividades plugadas com o *Scratch*, conforme descrito detalhadamente no capítulo 4.

Outra proposta de formação com professores foi apresentada por Bower et al. (2017), que utilizam *workshops*. Estes *workshops* ocorreram durante quatro módulos, onde foram discutidos temas como decomposição de problemas, reconhecimento de padrões, abstração e algoritmos.

Os autores realizaram a análise dos dados gerados pelos participantes utilizando o processo de agrupamento em categorias de códigos que foram obtidos por um processo de codificação inicial.

O processo de codificação utilizado nesta tese foi similar ao dos autores²², embora os mesmos utilizaram um *Software* para automatizar o processo e, nesta tese, foi realizado um processo manual. Nas conclusões, os autores apontaram como uma das limitações de sua pesquisa o tamanho da amostra de professores e o fato deste grupo avaliado ser específico da Austrália. Permitindo, assim, que outros pesquisadores pudessem criar pesquisas similares, bem como criar comparações com esta amostra.

Desta forma, em relação ao trabalho de Bower et al. (2017), considerou-se, nesta tese, que o grupo de professores amostrado foi complementar ao grupo apresentado na pesquisa dos autores, viabilizando a realização de algumas comparações com o trabalho citado.

No trabalho apresentado por Kalogiannakis e Papadakis (2017), os autores conduziram um estudo utilizando o *Scratch Jr*²³ com o objetivo de avaliar uma intervenção (*workshop*) com seis meses de duração, em um departamento de educação pré-escolar.

Os autores consideraram que os professores precisavam participar de seminários sobre o uso e a eficácia da mídia digital e procurar aplicativos que pudessem realmente promover diferentes habilidades cognitivas e de alfabetização. Assim, indicou-se que os professores da primeira infância precisavam de informações sobre o uso e as implicações de ferramentas tecnológicas.

O experimento associado ao trabalho dos autores²⁴ permitiu que alunos praticas-

²¹ Ibidem.

²² Ibidem.

²³ Que é baseado no *Scratch*, mas com foco em crianças mais jovens (5 aos 7 anos de idade): <https://www.scratchjr.org/>.

²⁴ Ibidem.

sem sequenciamento, raciocínio lógico e resolução de problemas. Assim como comportamentos positivos, tais como a colaboração e a comunicação.

Os resultados do trabalho revelaram que professores em formação tiveram aceitação positiva ao *Scratch Jr* em termos de utilidade e facilidade de uso. Então, a ferramenta seria apropriada na introdução aos conceitos básicos de programação e Pensamento Computacional, bem como o desenvolvimento de aplicações de professores do jardim de infância.

Outra pesquisa relacionada à formação de professores, e ao Pensamento Computacional, foi conduzida por [Yadav, Gretter e McLean \(2017\)](#) com mais de 130 professores em formação. Os autores buscaram entender como os professores sujeitos da pesquisa compreendiam e poderiam agregar o Pensamento Computacional em suas futuras aulas.

Os professores participantes responderam um questionário com questões abertas que geraram dados analisados por dois codificadores utilizando o *software NVivo*²⁵ em um processo de análise de conteúdo. Dentre os resultados obtidos pelos autores, foram geradas três conceituações principais pelos participantes sobre o Pensamento Computacional:

- O Pensamento Computacional poderia ser incorporado através da tecnologia, referindo-se à utilização do computador em sala de aula;
- O Pensamento Computacional poderia ser ensinado através da resolução de problemas ensinando como resolver os mesmos de uma forma lógica;
- O Pensamento Computacional poderia ser aplicado em áreas de conteúdo principal como a Matemática e estudos sociais.

O trabalho de [Romero, Lepage e Lille \(2017\)](#) trouxe a proposta de avaliar 120 estudantes de um curso de bacharelado em educação básica quanto à criatividade. Os autores utilizaram o *Scratch* para desenvolvimento de atividades, avaliando-as com o *Dr. Scratch*. Nesta tese, as duas ferramentas foram utilizadas da mesma forma, sendo o *Scratch* utilizado para criação de programas, e o *Dr. Scratch* utilizado para atribuir a pontuação aos programas.

A proposta de formação criada pelos autores²⁶ consistiu em apresentar aos participantes apenas dois recursos do *Scratch*: a criação de um novo ator com a possibilidade de arrastar e soltar blocos para criar o programa para o mesmo e o uso da bandeira verde para iniciar o programa. Posteriormente, os programas dos participantes foram enviados e avaliados e 91 participantes obtiveram um *score* total de 6 no *Dr. Scratch*.

²⁵ Nesta tese foi utilizado um processo similar de codificação sem a utilização de *Software*, criando os códigos manualmente.

²⁶ *Ibidem*.

Para verificar a criatividade contida nos programas, os autores adotaram como critério pontuar a utilização de blocos que não haviam sido ensinados aos participantes.

Já em Ramos e Espadeiro (2014), os autores analisaram os projetos *Scratch* de 44 professores, adotando a métrica proposta por Brennan e Resnick (2012) e as conclusões apontaram benefícios da introdução do Pensamento Computacional, auxiliando os professores a desenvolver capacidades de adaptação aos diferentes contextos de ensino e aprendizagem que podem enfrentar no futuro.

Ramos e Espadeiro²⁷ também destacaram a importância de apresentar propostas de trabalho cujos recursos tecnológicos pudessem ser integrados aos conteúdos e disciplinas com abordagens pedagógicas adequadas, similarmente ao que foi proposto por Paz (2017, p. 1657):

[...] é necessária uma formação inicial e continuada baseada nas transformações das práticas, identificando os saberes e competências necessárias a sua mobilização no trabalho, articulando teoria e prática, baseando-se na busca para resolução de problemas, que seja decidida e planejada em conjunto com esses profissionais. Visando preencher a lacuna da qualificação do professor regente, [...], com uma metodologia baseada no modelo andragônico, pode fornecer subsídios para que haja uma otimização do tempo e recursos gastos na formação desses profissionais.

A implementação de uma formação nos moldes propostos pelos trabalhos supracitados, foi apresentada por Eloy et al. (2017), que conduziram um projeto piloto com um processo metodológico considerando: implementação nas escolas, desenho do currículo, treinamento de professores, monitoramento e avaliação.

As ferramentas utilizadas para reunir informações foram reuniões com líderes locais, sessões presenciais e formulários virtuais de *feedback*.

Ainda, as ferramentas digitais utilizadas foram o *Code.org* e o *Scratch*. A formação dos professores utilizou materiais produzidos previamente em treinamentos do Instituto Ayrton Senna (IAS). Também seguindo a metodologia utilizada pelo IAS, foram propostas parcerias com as Secretarias Municipais de Educação.

Para garantir que a iniciativa seja liderada como uma política pública, dentro de um acordo formal as escolas interessadas designam os professores para participar do treinamento, organizando aulas como atividades extracurriculares.

Destaca-se que esta tese seguiu uma lógica de implementação metodológica muito similar àquela proposta por Eloy et al. (2017), tendo buscado uma parceria com a secretaria de educação do município onde a pesquisa foi realizada e considerando na metodologia uma etapa inicial de treinamento/formação dos professores e uma etapa posterior realizada nas escolas (implementação).

²⁷ Ibidem.

De uma forma geral, mesmo com todos os trabalhos apresentados, e a constatação de que há um movimento de pesquisadores e entidades que se preocupam com a necessidade de introduzir o Pensamento Computacional na sala de aula, há muito a ser feito.

Assim como pode ser verificado no trabalho de [Alencar \(2015, p. 1233\)](#), que relatou sobre os profissionais que estavam se preparando para atuar diretamente com o Pensamento Computacional, verificando que os mesmos ainda estavam aquém dos conhecimentos que seriam necessários:

Notou-se que os estudantes possuem clara limitação conceitual [...] tendo respostas fundamentadas em crenças, destoando do que de fato a literatura instrui. Existe pouca atividade por parte destes concluintes, na promoção do Pensamento Computacional com estudantes do ensino fundamental e médio. [...] Ao observar que as habilidades de uma turma de alunos concluintes estão carentes em seu fundamento basilar, acredita-se que tal cenário não foi estimado quando formularam os cursos de LCC no Brasil. Por tal, fazer uma profunda consideração sobre o que de fato espera-se de um profissional em LCC poderá ser determinante para a propagação do Pensamento Computacional, ou capaz deste marasmo ceifar o potencial cognitivo de uma geração que tende em atingir patamares mais sofisticados que as gerações contemporâneas.

Considerações sobre o Capítulo

Neste capítulo foram discutidos diversos aspectos referentes à formação de professores no contexto educacional, de uma forma geral, e também especificamente no contexto do Pensamento Computacional. Verificou-se, através da revisão da literatura apresentada, que a formação de professores foi um tópico amplamente discutido e de essencial importância para o ecossistema educacional.

A análise dos trabalhos utilizados demonstrou que ainda há muito a ser discutido sobre a melhor metodologia para formar professores, principalmente na formação continuada. Nossos currículos ainda são extremamente “engessados” e, conseqüentemente, se opõem e dificultam trabalhar de forma interdisciplinar e inovadora.

Portanto, pensar em implantar um contexto extremamente tecnológico como o do Pensamento Computacional, não é uma tarefa trivial. Principalmente se considerarmos que nossos professores não estão sendo formados em um ambiente que propicia o desenvolvimento de habilidades como a inovação e a utilização de tecnologia, o que resulta em dificuldade e resistência por parte dos professores quando precisam desenvolver estas habilidades.

Felizmente, nós, brasileiros, não estamos sozinhos nesta jornada, uma vez que a literatura apresentada traz exemplos de diversos outros países que também sinalizaram a urgência em discutir e implantar o Pensamento Computacional em sala de aula. Com

relatos de dificuldades similares às que encontramos em âmbito nacional. Assim, podemos utilizar estes exemplos para nortear nossas práticas.

4 Metodologia

Nesta seção, apresenta-se a metodologia utilizada para atender o Objetivo Geral proposto nesta tese: “Compreender como professores de Matemática e Informática, dos Anos Finais do ensino básico, se apropriam dos conhecimentos de um curso de formação em Pensamento Computacional, aplicando-os em atividades de sala de aula”.

Ainda, as atividades apresentadas na metodologia também permitem atender os seguintes objetivos específicos, já apresentados na Introdução desta tese:

- Avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional/*Scratch*, durante a etapa de formação;
- Avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional/*Scratch in loco* nas escolas, após a etapa de formação;
- Identificar as diferenças na percepção sobre a formação e nas atividades realizadas em sala de aula, de acordo com a formação do professor (Informática/Matemática);

A pesquisa foi estruturada em duas etapas: (i) Curso de formação - que ocorreu entre Setembro de 2017 e Outubro de 2018 e (ii) realização de atividades nas escolas - que ocorreu entre Maio e Dezembro de 2019.

O diagrama apresentado na figura 4 representa cada etapa realizada na metodologia, com as respectivas datas de execução, relacionando cada atividade realizada com o objetivo geral e com os objetivos específicos propostos na tese.

A etapa do curso de formação apresentada no lado esquerdo do diagrama consistiu de seis encontros presenciais com uma análise de conteúdo (que utilizou como dados de entrada os trechos de entrevistas realizadas com os professores) e análise dos programas gerados pelos professores. Esta etapa atendeu o objetivo específico de avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional e do *Scratch*.

A etapa de atividades nas escolas, apresentada no lado direito do diagrama, consistiu em sete visitas presenciais, com uma análise de conteúdo e análise dos programas gerados pelos professores e seus alunos. Esta etapa atendeu o objetivo específico de avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional *in loco* nas escolas.

As análises de conteúdo e análises dos programas, de ambas as etapas, atenderam o objetivo específico de identificar diferenças na percepção sobre a formação e nas atividades

realizadas em sala de aula, de acordo com a formação dos professores. E o conjunto destas etapas atendeu o objetivo geral da tese.

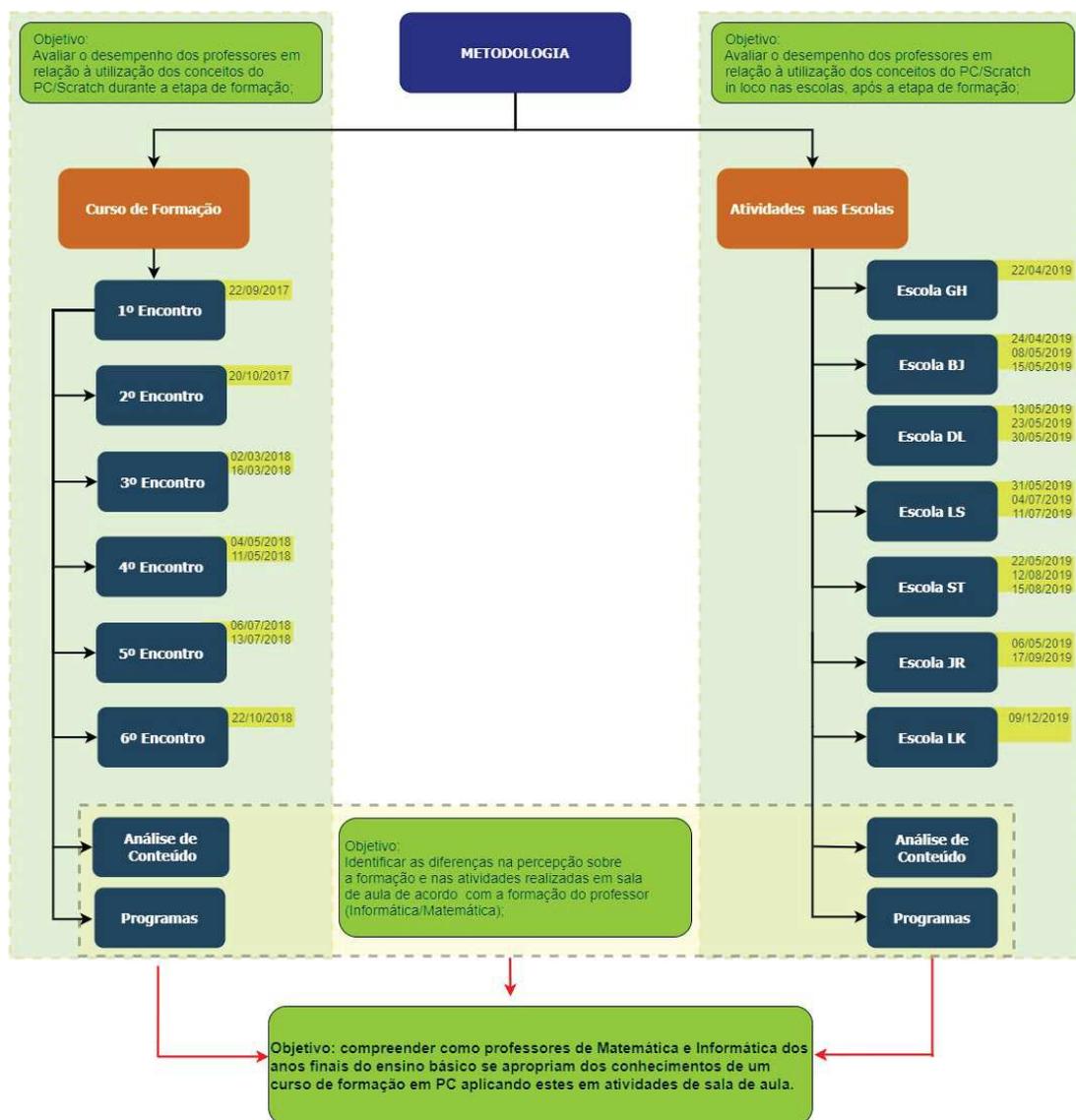


Figura 4 – Metodologia da tese.

Fonte: elaborada pelo autor.

Seguindo as recomendações para trabalhos futuros sobre ensino do Pensamento Computacional e formação de professores sugeridas em¹ Bower et al. (2017), esta tese atendeu os seguintes itens propostos pelos autores:

- Os autores sugerem que estudos realizados seis meses após as atividades de formação ajudariam a entender qual apoio poderia ser oferecido aos professores para que continuassem desenvolvendo seus conhecimentos.

¹ Cf. Seção 3.1.1.

Esta sugestão dos autores foi atendida nesta tese, uma vez que foram realizadas atividades com os professores nas escolas, aproximadamente sete meses após a conclusão da etapa de formação. Durante a realização destas atividades foi avaliado como os docentes propuseram utilizar sala de aula com os seus alunos, baseados nos conhecimentos adquiridos na formação. A análise dos dados gerados trouxe informações relevantes para a compreensão de como ajudar os professores nesta etapa.

- Os autores sugeriram investigar de que forma as capacidades dos professores seriam demonstradas em ambientes profissionais e práticas de sala de aula.

Esta sugestão dos autores foi atendida nesta tese, visto que foi realizada uma comparação da análise de conteúdo das falas dos professores e dos programas produzidos por eles em dois momentos distintos. Inicialmente, durante a etapa de formação, e posteriormente, na etapa de visitação às escolas, com os professores atuando em sala de aula com seus alunos. Também foi realizada análise dos programas desenvolvidos pelos alunos, permitindo, assim, verificar qual a influência do trabalho dos professores nestes programas.

4.1 O curso de formação

O primeiro passo do procedimento metodológico foi apresentar a proposta de pesquisa na Secretaria de Educação do município escolhido para aplicação da pesquisa².

Uma reunião foi agendada com a Secretária de Educação e a proposta de pesquisa foi contextualizada, com uma breve explanação sobre os conceitos do Pensamento Computacional e como estes poderiam ser utilizados como uma ferramenta em sala de aula.

Destacando que os conhecimentos relacionados ao Pensamento Computacional já eram um item discutido dentro da BNCC, e que preparar os professores, com relação a estes conhecimentos, era fundamental.

Foi estabelecida, então, uma parceria do pesquisador com a secretaria de educação, seguindo a ideia de aproximar a pesquisa com a política pública de maneira similar ao que foi proposto³ por [Eloy et al. \(2017\)](#).

A parceria foi benéfica para o pesquisador pois possibilitou a obtenção de dados de pesquisa, da mesma forma que foi positivo para a secretaria de educação pois, através da atuação do pesquisador, houve a disponibilização da formação continuada para os professores. A proposta foi vista como uma boa prática a ser desenvolvida junto aos professores do município.

² Por uma solicitação da Secretaria de Educação e como um alinhamento às questões éticas, não haverá referência ao município em questão nem aos nomes dos envolvidos.

³ Cf. Seção 3.1.1.

Após os acertos necessários com a secretaria, foi agendada uma reunião com a coordenadora do departamento de educação e com o Chefe da divisão administrativa e pedagógica das Escolas Municipais de Ensino Fundamental (EMEFs).

O objetivo da reunião foi verificar qual grupo de professores que iria participar da formação e qual a logística necessária, durante o período de formação, para adequar os horários disponibilizados na formação, com os horários disponíveis dos professores.

A proposta inicial da secretaria foi disponibilizar a formação somente para os professores de Informática, dado que a base de conhecimento destes era mais alinhada ao curso de formação proposto, e assim teriam maior predisposição para interagir com a ferramenta *Scratch* e os conceitos a serem abordados.

O pesquisador questionou a possibilidade de também integrar na formação os professores de Matemática, destacando que, conforme a [BNCC \(2019\)](#), a disseminação dos conhecimentos relativos ao Pensamento Computacional cabe aos docentes daquela disciplina.

A ideia de contemplar inclusive os docentes da Matemática foi bem vinda para a secretaria e alinou-se à proposta de pesquisa. Destaca-se, ademais, que a Matemática tem forte relação com o Pensamento Computacional e conhecimentos prévios desta podem contribuir para assimilar conhecimentos de programação ([BARCELOS; SILVEIRA, 2012](#); [BARCELOS; SILVEIRA, 2013](#)). Uma explicação sobre como o Pensamento Computacional se encaixa na Matemática pode ser verificada em [Weintrop et al. \(2016\)](#), que destacaram quais são os requisitos dos professores que vão formar os alunos para atuarem nestes campos.

Com a proposta apresentada e aprovada pela Secretaria de Educação, foi enviado um memorando para os(as) diretores(as) das EMEFs com um convite formal aos professores das disciplinas de Matemática e de Informática. Um total de 25 EMEFs foram notificadas sobre a formação.

Houve convite para que os professores participassem do curso de formação. Após as respostas das escolas, um total de 53 professores (13 da disciplina de Informática e 40 da disciplina de Matemática) tiveram participação confirmada. Porém, no decorrer do curso alguns professores não conseguiram continuar participando e assim as atividades finalizaram com a participação de 49 docentes.

O curso de formação, proposto pelo pesquisador, previa uma abordagem plugada utilizando o *Scratch* para explorar os conceitos do Pensamento Computacional. Consequentemente, buscou-se um local com um número adequado de computadores para atender à quantidade de professores inscritos.

Foi realizada, então, uma parceria junto com um Centro de Formação Profissional (CFP), do SENAI, cujo diretor do CFP disponibilizou uma sala de aula com 32 compu-

tadores com acesso à *internet* para utilização pelo pesquisador durante a realização do curso.

A utilização do *Scratch*, associada à disponibilidade do laboratório com computadores, configurou recursos tecnológicos que poderiam ser posteriormente integrados aos conteúdos ministrados pelos professores, similarmente ao que foi recomendado por⁴ Ramos e Espadeiro (2014).

Com relação à distribuição dos professores em sala de aula, foi decidido dividir a turma em dois subgrupos que participaram da formação ou no turno da manhã, ou no turno da tarde, atendendo assim: (1) às necessidades de horários apresentadas pelos professores e (2) evitar o impacto negativo que um número grande de alunos em sala de aula pode gerar, assim como constatado por Monks e Schmidt (2011).

A ideia central do curso foi formar os professores para que eles pudessem trabalhar questões elementares sobre o Pensamento Computacional com seus estudantes, possibilitando, do mesmo modo, desenvolver programas no *Scratch*, contemplando diferentes níveis das dimensões⁵ do Pensamento Computacional.

As atividades desenvolvidas em cada um dos encontros⁶ foram pensadas e desenvolvidas para serem atividades simplificadas, de fácil compreensão e contextualizadas de maneira alinhada à realidade dos docentes participantes.

O intuito de desenvolver atividades alinhadas à realidade dos docentes buscou diminuir a rejeição à formação e evitar que o docente pudesse sentir-se “obsoleto, deslocado [...] por não conseguir dominar os conceitos aprendidos”, conforme citado por Paz (2017, p. 1660) ao referenciar cursos de formação que não tratam a tecnologia como um meio, mas um fim.

O formato do curso contemplou atividades presenciais e à distância, enquadrando-se, assim, de acordo com as atividades propostas por (Digital Promise, 2017)⁷. Nas atividades presenciais foi priorizada a utilização do *Scratch*, ferramenta com a qual os professores conseguiram praticar os conceitos do Pensamento Computacional discutidos, com um enfoque maior para a utilização da ferramenta (*Scratch*) como recurso de sala de aula.

Ao contrário de situações onde cursos de formação priorizam ferramentas voltadas à organização da aula pelo docente, tal como discutiram Schuhmacher, Filho e Schuhmacher (2017, p. 575)⁸.

A proposta de atividades para o curso levou em consideração os conceitos do Pensamento Computacional de forma similar à análise de Ramos e Espadeiro (2014), com

⁴ Cf. seção 3.1.1.

⁵ Apresentadas na Tabela 1 do Capítulo 2.

⁶ Estas atividades foram descritas no anexo B.

⁷ Cf. seção 3.1.

⁸ Cf. Introdução do Capítulo 3.

as seguintes atividades:

- Criação de usuário e senha e como utilizar os comandos para criar um programa no *Scratch*;
- Comandos disponíveis no *Scratch* e como relacionar os mesmos com conteúdos desenvolvidos em sala de aula;
- Pensamento Lógico: estruturas de decisão se-então e se-então-senão;
- Interatividade com o usuário: utilizando Bandeira Verde para iniciar o programa, entrada de dados pelo teclado e perguntas ao usuário;
- Repetir até que, e repetir sempre;
- Representação de dados: modificando propriedades de objetos. Utilizando variáveis.
- Paralelismo: múltiplos atores e múltiplos *scripts*.

Para associar comandos do *Scratch*, com os conceitos do Pensamento Computacional, foram utilizados os seguintes exemplos como base:

- Abstração: permite desconsiderar detalhes não fundamentais e concentrar-se na solução de um problema. Como exemplo no *Scratch*, vamos imaginar que queremos descrever uma determinada trajetória de um ator na tela.

O ator pode ser qualquer objeto, desde um animal, personagem, número, letra, dentre outros. Independentemente do ator utilizado, estamos dedicados a dar movimento ao mesmo, o que vai ser realizado facilmente com o comando “move”.

Detalhes como a forma que o ator se movimenta, qual o movimento esperado das pernas ou patas, ou determinada parte do corpo ou objeto, não foram discutidas (por mais que possa ser criada uma animação para movimentar as pernas de um ator, por exemplo), o objetivo foi movimentar o ator;

- Decomposição de problemas: como resolver um problema grande e complexo? Vamos transformá-lo em pequenos problemas mais simples e, conseqüentemente, mais fáceis de solucionar. Um exemplo voltado para a Matemática é o de calcular o valor de uma função. Digamos $y = x^2 + 2$. Esta função pode ser considerada como dois pequenos cálculos: (1) a exponenciação x^2 , que neste caso vai ser executada por $x * x^9$,¹⁰ e (2) a soma deste resultado com a constante 2.

⁹ Não há um comando direto de exponenciação para bases diferentes de 10 e de e (número de *Neper*, número de *Euler* ou número *Neperiano* dentre outras denominações).

¹⁰ Observar também que o símbolo “=” não aparece na função criada, sendo substituído pelo comando “mude y para”, que faz com que o valor calculado seja atribuído à variável y . O símbolo = no *Scratch* é utilizado para comparar se dois elementos são iguais, sejam estes números, caracteres ou *Strings*.

No *Scratch*, para trabalharmos com este exemplo, foram utilizados operadores e variáveis¹¹. A figura 5 mostra a função criada e os blocos que a compõem separadamente.

Em um estudo conduzido por Selby (2015), observou-se que a decomposição foi o conceito que apresentou maior dificuldade de assimilação no contexto do Pensamento Computacional. Logo, trabalhar com um conceito matemático simples, como calcular uma função decompondo a mesma em pequenos fragmentos, buscou promover a compreensão do conceito de decomposição.

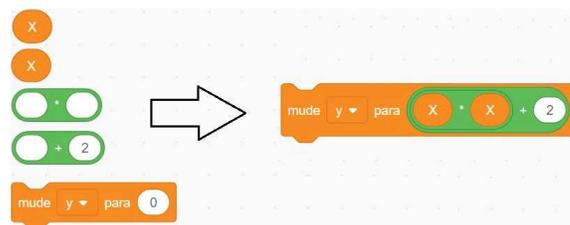


Figura 5 – Função $y = x^2 + 2$ implementada no *Scratch*.

Fonte: elaborada pelo autor.

- Pensamento Lógico: é o processo de utilizar regras lógicas para resolver problemas, como, por exemplo,¹² podemos utilizar o condicional SE para tomar uma decisão: movimentar um ator no *Scratch* utilizando o *mouse*, e, com base na coordenada horizontal e/ou vertical do ator, trocar a cor do mesmo.

Este exemplo poderia utilizar também a estrutura SE - SENÃO, mas a utilização de duas estruturas SE foram adotadas para facilitar a compreensão da lógica.

Destaca-se também a integração de outros comandos, como foi o caso da utilização do comando de controle “Sempre”, utilizado para garantir que o usuário pudesse movimentar o ator continuamente após iniciar a execução do programa e do comando de troca de fantasias, adotado para mudar a cor do ator. A figura 6 destacou os blocos utilizados para implementar a lógica proposta.

- Sincronização: a sincronização permite que um “pedaço” do programa só seja executado após a ocorrência ou finalização de um determinado evento ou de outro bloco de programa. Uma forma de sincronizar no *Scratch* consiste na utilização de mensagens.

No exemplo¹³ da figura 7, o besouro vai começar a se movimentar quando o gato passar da posição horizontal 50 e enviar a mensagem 1.

¹¹ Este exemplo foi posteriormente retrabalhado para demonstrar a utilização da interatividade com o usuário, como os comandos “pergunte” e “diga”, fazendo com que o usuário pudesse informar o valor desejado para uma variável.

¹² Disponibilizado em: <https://scratch.mit.edu/projects/284387217>.

¹³ Disponibilizado em <https://scratch.mit.edu/projects/284956697>.



Figura 6 – Estrutura condicional “SE - ENTÃO” no *Scratch*

Fonte: elaborada pelo autor.

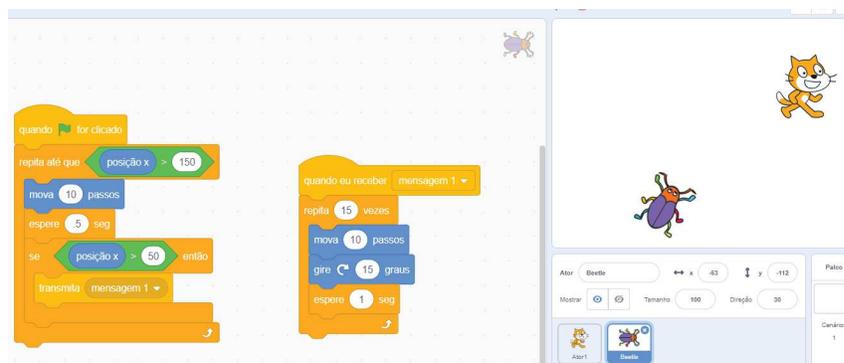


Figura 7 – Sincronização implementada no *Scratch*.

Fonte: elaborada pelo autor.

- Paralelismo: O paralelismo ocorre quando mais de um evento entra em execução simultaneamente. No *Scratch* podemos criar, por exemplo, um programa onde dois (ou mais) atores interagem, assim como o exemplo¹⁴ representado na figura 8, onde o cachorro e a bola ficavam se movimentando na tela até que cada um tocasse no outro, com um *Script* de programação para cada ator.
- Noção algorítmica de controle de fluxo: Um algoritmo é um processo passo a passo que resolve um problema ou conclui uma tarefa. Pode servir tanto para resolver um problema complexo de inteligência artificial, como para descrever uma simples tarefa.

Um simples algoritmo para animar um ator foi proposto no *Scratch*: (1) desenhe sua trajetória com uma caneta; (2) mova 10 passos para frente; (3) gire 90°; (4) espere 1 segundo; (5) repita 4 vezes.

O resultado deste algoritmo foi o desenho de um quadrado, conforme mostra a figura 9, com o código criado¹⁵ no *Scratch* seguindo o algoritmo proposto.

¹⁴ Disponibilizado em <https://scratch.mit.edu/projects/220590755>.

¹⁵ Disponibilizado em: <https://scratch.mit.edu/projects/284707863>.

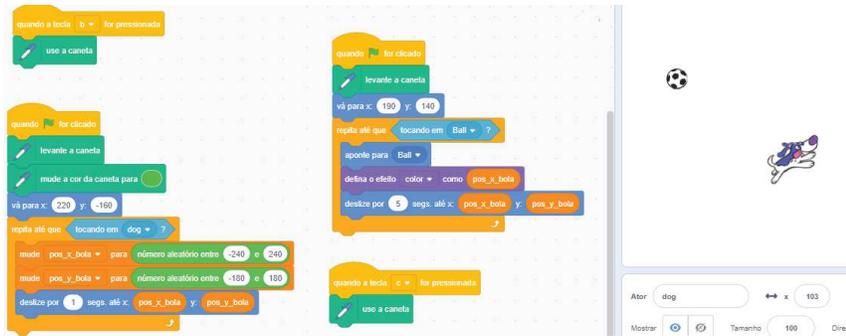


Figura 8 – Paralelismo implementado no *Scratch*.

Fonte: elaborada pelo autor.

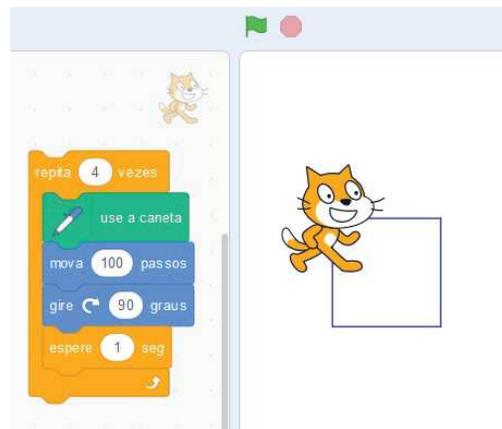


Figura 9 – Algoritmo implementado no *Scratch*.

Fonte: elaborada pelo autor.

- Interação com usuário: ocorre sempre que o usuário interagir de alguma forma com o computador ou com um programa que estiver sendo executado. Uma forma de interação é a utilização do *mouse* ou teclado, que são equipamentos que permitem ao ser humano inserir dados no computador.

No exemplo da decomposição representado na figura 5, ocorreu interação com o usuário quando o programa “perguntava” ao usuário qual era o valor desejado para a variável x .

- Representação de dados: dados podem ser armazenados e tratados de diferentes formas. No *Scratch*, a forma de trabalhar com dados é através da utilização de variáveis e listas. No exemplo sobre decomposição, representado na figura 5, foram utilizadas duas variáveis: x e y .

Para cada um dos encontros previstos foi contextualizado um ou mais conceitos do Pensamento Computacional, associando o(s) mesmo(s) com os comandos do *Scratch* que

poderiam implementá-lo(s), relacionando, também, o programa criado com um conteúdo da Matemática e/ou Informática.

Assim, considerou-se que foi utilizado um modelo Pedagógico Relacional similar ao proposto por Becker (1999), onde foi apresentado aos professores um material com significado para os mesmos e com o qual foi possível que construíssem o seu próprio conhecimento.

De forma a garantir que todos os comandos utilizados, durante o curso, possuíssem uma fonte de referência adicional utilizado no documento¹⁶ disponibilizado pelo *Scratch* Brasil (<http://www.scratchbrasil.net.br/>) e as “cartas”¹⁷ *Scratch* que trazem uma referência rápida e visual aos comandos básicos.

Adicionalmente, para cada conceito do Pensamento Computacional e comando do *Scratch* que foram discutidos durante a formação, também foram desenvolvidos exemplos que pudessem validar a proposta. No decorrer do curso, os professores tiveram a oportunidade de criar programas no *Scratch* para testar cada um dos conceitos do Pensamento Computacional que foram discutidos, bem como conhecer os blocos de programação disponíveis no *Scratch*.

Com o intuito de gerar dados para avaliar os professores, foi solicitado aos mesmos que criassem três programas, respectivamente: (1) no início da formação, (2) durante a formação e (3) no final da formação.

Deste modo, foi possível realizar uma comparação entre diferentes momentos da formação para verificar a forma como o *Scratch* e os conceitos do Pensamento Computacional foram utilizados.

O curso de formação em Pensamento Computacional foi dividido em seis encontros presenciais que ocorreram entre 22 de setembro de 2017 e 22 de outubro de 2018. Estes eventos estão relatados nas próximas seções.

4.1.1 Primeiro encontro presencial em 22/09/2017

Após os convites enviados para as escolas, dos 53 professores confirmados para participar da formação, 51 compareceram ao primeiro encontro presencial.

Neste primeiro encontro, o pesquisador fez uma apresentação formal sobre o Pensamento Computacional explicando a proposta e o objetivo da pesquisa, destacando a importância da introdução do Pensamento Computacional no cenário educacional e o alinhamento desta proposta com a BNCC.

Os professores participantes assinaram um Termo de Consentimento Livre e Es-

¹⁶ <https://bit.ly/MaterialScratch>, acessado em 12/09/2017.

¹⁷ <https://bit.ly/2Aj7cD2>, acessado em 12/09/2017.

clarecido (apresentado no Apêndice A), tomando ciência da utilização dos dados gerados durante a formação para publicações acadêmicas e que seria garantido o anonimato¹⁸ sobre a identidade de todos os participantes.

As ferramentas computacionais propostas para serem utilizadas, durante a formação, foram apresentadas aos professores sendo: (a) o *Scratch*: utilizado para desenvolver as atividades relacionadas as dimensões do Pensamento Computacional (como Abstração e Pensamento Lógico); e (b) o *GOOGLE Classroom*: para interação em grupo dos participantes com o pesquisador e envio de tarefas. Além disso, foi proposta a criação de um grupo de *Whatsapp* para agilizar a comunicação entre os participantes e o pesquisador.

Com relação ao *Scratch*, os professores receberam informações sobre como utilizá-lo nos modos *offline*¹⁹ e também o modo *online*, acessando <https://scratch.mit.edu/>, a partir da criação de um usuário para poder criar os programas e compartilhá-los.

O encontro foi dimensionado para 4 horas de duração e dividido nos seguintes momentos:

- Recepção dos professores;
- Apresentação dos professores e do pesquisador;
- Aplicação de questionário sobre intenção de utilização do computador (discutido na seção 4.1.2.1);
- Motivação sobre o tema do Pensamento Computacional;
- Apresentação do *Scratch*;
- Atividades a serem desenvolvidas entre o primeiro e o segundo encontros;
- Encerramento.

No grupo de professores, especificamente com relação aos 13 professores de Informática: dois deles não conheciam o *Scratch*; três já utilizavam em sala de aula com os alunos; e os demais professores relataram que já haviam conhecido o *Scratch* mas, ou não o haviam utilizado, ou utilizaram-no sem aprofundamento.

No que diz respeito aos professores de Matemática, de forma geral, eles afirmaram não conhecer o *Scratch*.

¹⁸ Sempre que necessário citar um professor no decorrer deste trabalho foi utilizada uma codificação com a letra P seguida por dois números, por exemplo P10.

¹⁹ Quando o *Scratch* é instalado no computador atendendo os casos de escolas sem acesso à *internet*

O primeiro passo para apresentar o *Scratch* foi acessá-lo *online*, onde os professores foram instruídos a criar um usuário e senha²⁰, porém, verificou-se posteriormente que alguns professores acabaram por esquecer os seus usuários e/ou senhas, dificultando a condução de atividades posteriores.

Foi criado um programa simples, representado na figura 10, contendo somente um comando de “Movimento” que permitiu explicar os conceitos básicos relativos ao *Scratch*.

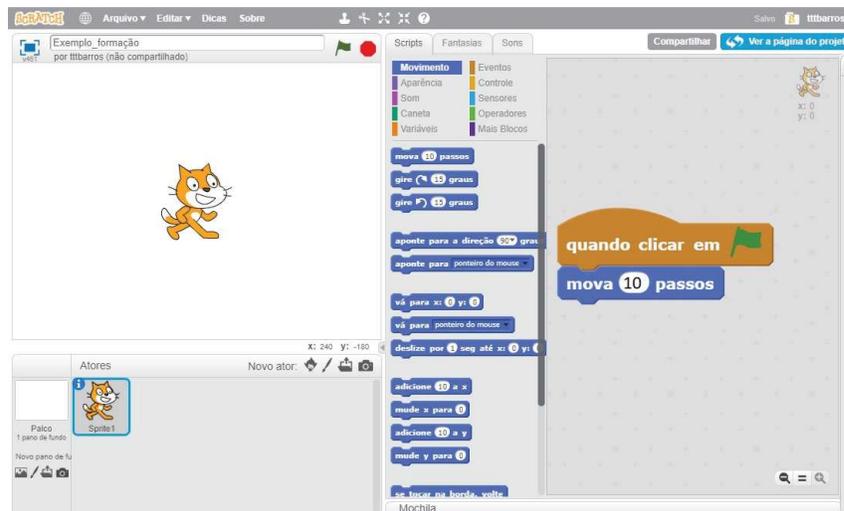


Figura 10 – Programa exemplo com comando de movimento implementado no *Scratch*.

Fonte: elaborada pelo autor.

O programa representado na figura 10 permitiu explicar os seguintes conceitos:

- Palco: tela onde os atores ficam dispostos. O palco pode receber uma imagem auxiliar para criar um cenário para o programa criado. Por exemplo: pode ser utilizada uma imagem do fundo do mar para a criação de uma história sobre/jogo sobre sereias;
- Ator: cada elemento gráfico disposto sobre o(s) palco(s). O ator padrão utilizado é o gato com a cor laranja, símbolo do *Scratch*;
- Fantasias: cada uma das diferentes imagens associadas com um ator. Por exemplo: o gato pode ser pintado de outra cor, ou utilizar um acessório para compor uma fantasia;
- Comandos: elementos que são associados para criar um *script*, que é a forma de escrever um programa no *Scratch*. Estes comandos foram divididos em 10 grupos distinguíveis por cores;

²⁰ Como recomendação a outros pesquisadores que realizarem uma formação similar, recomenda-se que a criação de usuários e senhas fiquem por conta do pesquisador evitando possíveis complicações por esquecimento destes dados por parte dos sujeitos da pesquisa.

- Como criar, salvar e compartilhar um projeto criado no *Scratch*. Também foi verificado como utilizar programas criados por outros usuários e “remixar” os mesmos, ou seja, utilizar um programa já criado e modificá-lo.

Após o contato inicial com o *Scratch*, foi discutido com os professores sobre as bases e conceitos do Pensamento Computacional, para esclarecer a importância dos mesmos e contextualizar de que forma cada um pode ser trabalhado, fazendo uma associação com tópicos das disciplinas de Matemática/Informática.

Por exemplo: para se discutir sobre o Pensamento Lógico, ou Raciocínio Lógico²¹, foram utilizados os operadores relacionais disponíveis no *Scratch*: maior que (“>”), menor que (“<”) e igual (“=”). O programa criado²² para exemplificar este contexto está representado na figura 11.



Figura 11 – Programa exemplo para discussão sobre o conceito de raciocínio lógico.

Fonte: elaborada pelo autor.

A ideia central do programa foi a utilização dos operadores relacionais, os quais fazem parte do conceito relativo à lógica. Quando o usuário informava sua idade, utilizando o *prompt* para entrada de dados, o número informado era armazenado na variável resposta.

Então, a variável resposta era comparada com o valor de referência (18). E “SE” o valor de entrada fosse maior ou igual a 18, o ator informava que o usuário era maior de idade. Caso contrário (“SENÃO”), informava que o usuário era menor de idade.

Ainda que este exemplo seja classificado como básico (6 pontos possíveis de 21) na análise do *Dr. Scratch* (figura 12), uma série de considerações puderam ser efetuadas:

- Foram utilizados sete comandos do *Scratch*, sendo eles: “quando clicar em”, “pergunte e espere a resposta”, “se/senão”, “relacionais maior que (“>”) e igual (“=”), operador lógico “ou” e comando de aparência “diga”;
- Como não existe o comando relacional maior ou igual que (“>=”), foi necessário utilizar o operador lógico “ou”, para unir os dois operadores existentes. Estes 3

²¹ Assim como descrição da Tabela 1 do Capítulo 2.

²² Disponibilizado em: <https://scratch.mit.edu/projects/277713847/>.

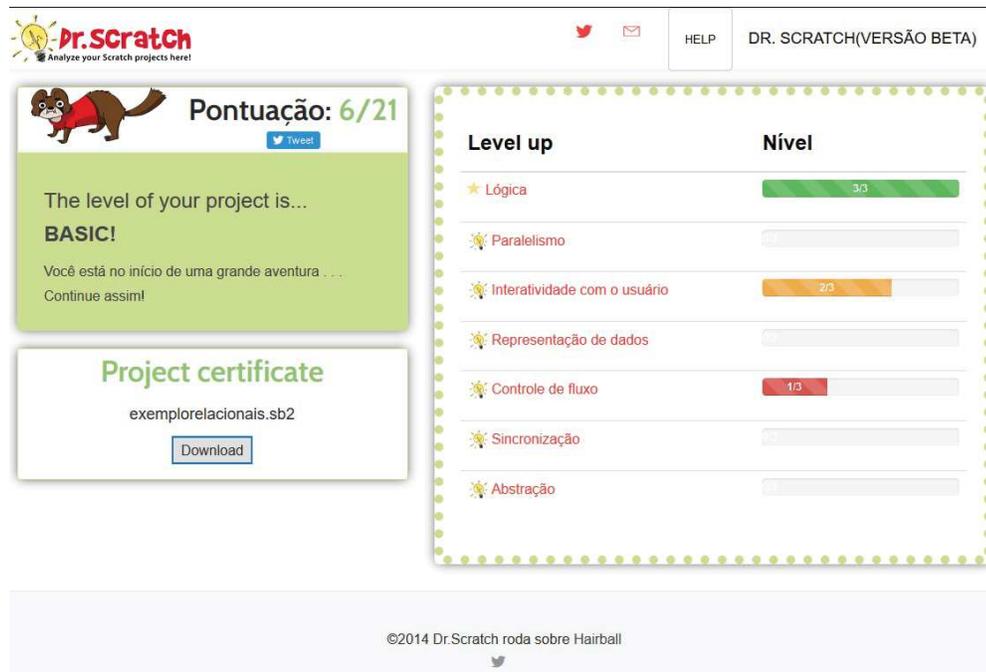


Figura 12 – Pontuação do *Dr. Scratch* para o programa da figura 11.

Fonte: elaborada pelo autor.

operadores são os 3 itens contemplados no item “Lógica” da pontuação do *Dr. Scratch* na figura 12.

- Para o programa ser executado, ou “rodar”, foi utilizado o comando “quando clicar em”, que, graficamente, representa o clique na bandeira verde que indica a execução do programa. Este comando pertence à Interatividade com o usuário, junto com o comando “diga”, que obteve 2 de 3 pontos possíveis;
- Para permitir que o programa possa verificar se o valor de entrada informado era maior ou menor que 18, foi utilizado um comando/estrutura condicional se/senão, o qual pontuou como controle de fluxo.
- O programa trouxe, de forma subliminar, o conceito de variável na utilização do comando “resposta”, que armazenava o número informado pelo usuário para posterior comparação;
- Com os elementos apresentados neste programa, foi possível demonstrar para os participantes que outros programas poderiam ser derivados deste. E criar, por exemplo, um programa para somar um número dado pelo usuário com um número informado pelo computador. Utilizando somente mais um comando novo para gerar um número aleatório.

O intervalo de tempo ocorrido entre o primeiro e o segundo encontros presenciais foi de 28 dias e que, neste período as ferramentas propostas para comunicação, foram utilizadas para enviar dicas aos professores e prepará-los para o restante do período de formação.

A primeira atividade postada no ambiente *GOOGLE Classroom* foi um convite para que os professores realizassem o teste de um jogo criado no *Scratch* pelo pesquisador, com o intuito de mostrar as possibilidades de utilização do *Scratch* e de alguns blocos básicos de programação do mesmo.

Contudo, nesta primeira atividade, ocorreram apenas três acessos (figura 13a) realizados pelos 42 professores que criaram usuários e se cadastraram na sala de aula virtual.

Posteriormente foi disponibilizado um conjunto de arquivos para promover a interação dos professores com o *Scratch*, cujos arquivos consistiram em um instalador da versão *offline* do *Scratch*²³, um tutorial de Lógica Básica do *Scratch*²⁴ e um material com cartas²⁵ contendo dicas sobre os blocos de programação e características da IDE do *Scratch*.

E, novamente, ocorreram poucos acessos ao material postado, com apenas cinco professores tendo acessado os arquivos que foram disponibilizados, conforme apresentado na figura 13b.

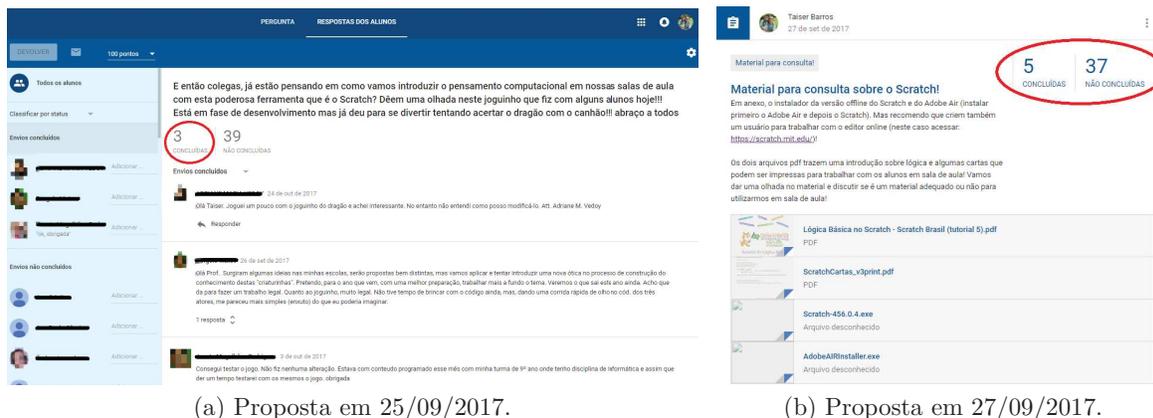


Figura 13 – Número de acessos às atividades.

Fonte: elaborada pelo autor.

A última tarefa que foi criada utilizando o *GOOGLE Classroom*, antecedendo o

²³ Junto com o *Adobe Air* que é necessário para funcionamento do *Scratch*.

²⁴ Material reorganizado e adaptado por Suellen Rodolfo Martinelli, disponibilizado em www.scratchbrasil.net.br.

²⁵ Adaptadas e traduzidas por Christiane Gresse von Wangenheim, Vinicius Rodrigues Nunes, Giovane Daniel dos Santos e Juliane Vargas Nunes a partir das cartas *Scratch* (<http://scratched.media.mit.edu/resources/scratch-cards>).

segundo encontro presencial, buscou estimular os professores na criação de uma proposta de atividade relacionada aos conteúdos desenvolvidos em sala de aula e que utilizasse o *Scratch* para implementá-la. O enunciado da tarefa foi:

O que estou ensinado e como poderia utilizar o Scratch para aprimorar? Pensando em qual conteúdo você está desenvolvendo em sala de aula com os seus alunos (seja na área da Matemática ou da Informática), tente criar uma situação onde a utilização do Scratch possa agregar à sua atividade docente. Por exemplo, os professores de Matemática que estão ensinando frações! Que tal criarmos uma animação que relaciona número de personagens/objetos exibidos na tela com os numeradores e denominadores das frações??? Sua tarefa é propor aqui uma atividade, problema, dúvida, ..., que você julgue passível de ser respondida ou desenvolvida utilizando o *Scratch*! Depois das postagens vamos discutir suas ideias e propor implementações! Para aqueles que tiverem alguma ideia que possa virar um programa, a comunidade²⁶ pode ajudar na implementação! Não deixe de participar, pois sua opinião é fundamental para seu desenvolvimento, de seus colegas e de nossa comunidade!!!

Com relação à última tarefa solicitada, foram registradas vinte e duas entregas que foram discutidas em aula com o grupo de professores. Por isso, as ideias trazidas por eles podiam ser utilizadas pelo grupo para gerar propostas de programas e estimular a troca de conhecimento entre os professores.

4.1.2 Segundo encontro presencial em 20/10/2017

Da mesma forma que ocorreu no primeiro encontro, o roteiro proposto para o segundo encontro também foi dimensionado para 4 horas de duração e dividido nos seguintes momentos:

- Recepção dos professores;
- Aplicação de questionário sobre intenção de utilização do computador (discutido na seção 4.1.2.1);
- Diálogo sobre as propostas de implementação com o *Scratch*;
- Comandos do *Scratch* que podem ser utilizados para implementar as propostas;
- Entrevista;
- Encerramento.

²⁶ Pretendia-se criar com os professores uma Comunidade de Prática (CP): grupo de indivíduos com distintos conhecimentos que compartilham interesse sobre um tópico aprofundando seus conhecimentos através da participação ativa em processos de colaboração (WENGER, 1999). Porém, não foram identificados colaboração e interação entre os professores nas atividades propostas, sendo estes, fatores essenciais (FERREIRA; SILVA, 2014) de uma CP. Assim a ideia foi desconsiderada.

Durante o segundo encontro, os professores relataram como estavam (ou não estavam) interagindo com as ferramentas *online* disponibilizadas, quais as ideias que surgiram com relação à possibilidade de utilizar os conceitos discutidos desde o primeiro encontro junto aos seus alunos, e como a formação poderia contribuir em suas atividades docentes.

Com relação aos professores que efetivamente conseguiram produzir resultados utilizando o *Scratch*, destacou-se a criação de um programa para introduzir o conceito de exponenciação criado com a participação da professora de Matemática P17 (que trouxe a necessidade em relação ao conteúdo que estava sendo desenvolvido na disciplina) e do professor de informática P16 que auxiliou na implementação do programa²⁷.

Ainda, o professor P16 trouxe a proposta de um programa que calculava o quadrado de um número informado, exibindo uma quantidade correspondente de elementos no palco²⁸.

Também foi discutida a opinião dos professores sobre a utilização do ambiente *GOOGLE Classroom* para entregar as tarefas, uma vez que o número de acessos ao ambiente havia sido pequeno.

Embora tenha havido relatos de dificuldades em utilizar o ambiente, o grupo de professores entrou em consenso com relação a usar somente o grupo de *Whatsapp* para comunicação e envio de tarefas.

4.1.2.1 Questionário sobre intenção de utilização do computador

Uma vez que as atividades da formação foram realizadas no *Scratch* e, consequentemente, precisavam ser desenvolvidas utilizando um computador, adotou-se um questionário sobre intenção do uso do computador como ferramenta em sala de aula proposto por Ma, Andersson e Streith (2005). As perguntas do questionário foram adaptadas com livre tradução e apresentadas no apêndice C.

Para cada uma das perguntas do questionário aplicado, os professores responderam utilizando uma escala *Likert* de sete²⁹ pontos, variando desde “concordo totalmente” a “discordo totalmente”.

A coleta das respostas foi efetuada por meio do ambiente de formulários do *GOOGLE*. Ao todo, 44 professores participaram.

Os resultados do questionário aplicado foram analisados utilizando como parâmetro o Alfa de *Cronbach* (CRONBACH, 1951), o qual estimou a confiabilidade do questionário.

²⁷ Disponibilizado em: <https://scratch.mit.edu/projects/181041709>.

²⁸ Disponibilizado em: <https://scratch.mit.edu/projects/177273505>.

²⁹ A utilização da escala de sete pontos foi fundamentada pelos estudos de Júnior e Costa (2014) indicando que em escalas de múltiplos itens a confiabilidade é maior utilizando-se sete ou mais pontos e menor quando os itens possuem menos de 5 pontos.

nário medindo-se a correlação entre todas as respostas e calculando α a partir da variância dos itens individuais e soma dos itens de cada avaliador referente a todos os itens.

A consistência interna dos dados pode ser avaliada segundo o valor de α , conforme valores propostos por George e Mallery (2007), apresentados na Tabela 2.

Tabela 2 – Consistência interna do questionário conforme valores de α .

Valor de α	Consistência interna
0,91 ou mais	Excelente
0,90 - 0,81	Bom
0,81 - 0,71	Aceitável
0,71 - 0,61	Questionável
0,61 - 0,51	Inaceitável

Fonte: adaptado de George e Mallery (2007).

Para o questionário do primeiro encontro, obteve-se um α de 0,961 e para o questionário do segundo encontro um α de 0,960, o que permitiu afirmar que a confiabilidade dos dados foi excelente.

Para verificar se houve variação significativa na intenção de utilização do computador por parte dos professores, entre a primeira e a segunda aplicação do questionário foi utilizado o teste t de *student*, buscando comparar as médias de cada item de medição respondidos nos questionários para verificar se, de alguma forma, a formação oferecida aos professores influenciou a intenção dos mesmos com relação à utilização do computador.

Logo, conforme apresentado na tabela 3, verificou-se que houve variação significativa (nível de significância de 0,95) para os itens 7 e 10 do questionário, o que permitiu afirmar que a formação pode ter influenciado a maneira como os professores veem o potencial dos computadores para auxiliá-los em seu trabalho (item 7) e na sua autoconfiança para utilizá-lo sozinhos (item 10). Porém, globalmente, não foi possível afirmar que a formação provocou mudanças na intenção dos professores em utilizar o computador como ferramenta em sala de aula.

4.1.2.2 Entrevista

Após o primeiro encontro, como o número de entregas das atividades propostas foi considerado pequeno, o pesquisador realizou uma entrevista semiestruturada com doze professores voluntários, buscando elucidar quais dificuldades foram detectadas na implementação.

A entrevista semiestruturada tem por característica permitir que os entrevistados possam falar livremente sobre assuntos que surjam durante os questionamentos criados pelo pesquisador, cujo enfoque se volta em explorar um assunto que esteja sendo estudado (GERHARDT; SILVEIRA, 2009).

Tabela 3 – Resultados do teste t para as duas aplicações do questionário.

Pares comparados	t	σ
Par 01 1E01 - 2E01	-0,161	0,873
Par 02 1E02 - 2E02	1,555	0,129
Par 03 1E03 - 2E03	0	1
Par 04 1E04 - 2E04	1,186	0,244
Par 05 1E05 - 2E05	1,146	0,26
Par 06 1E06 - 2E06	1,886	0,068
Par 07 1E07 - 2E07	2,547	0,016
Par 08 1E08 - 2E08	0,745	0,462
Par 09 1E09 - 2E09	0,147	0,884
Par 10 1E10 - 2E10	2,24	0,032
Par 11 1E11 - 2E11	1	0,324
Par 12 1E12 - 2E12	0,172	0,865
Par 13 1E13 - 2E13	0	1
Par 14 1E14 - 2E14	1,152	0,257
Par 15 1E15 - 2E15	-0,106	0,916
Par 16 1E16 - 2E16	0,089	0,93
Par 17 1E17 - 2E17	0,592	0,557
Par 18 1E18 - 2E18	0,415	0,681
Par 19 1E19 - 2E19	0,706	0,485
Par 20 1E20 - 2E20	1,313	0,198
Par 21 1E21 - 2E21	1,39	0,174

Para iniciar a entrevista com os professores voluntários, o questionamento do pesquisador foi: “Você considera relevante para sua atividade docente o contexto do Pensamento Computacional e utilização do *Scratch*? Você teve alguma dificuldade em tentar desenvolver atividades junto aos alunos, ou não possuiu acesso a laboratórios e equipamentos? Comente sobre sua real situação em sua escola frente a esta capacitação para implementar atividades relativas ao Pensamento Computacional”.

Alguns trechos³⁰ da entrevista - considerados pertinentes para elucidar possíveis fatores que explicassem a pequena participação do grupo nas atividades propostas -, foram destacados a seguir:

- P09: "Considero muito importante esta capacitação, mas tenho muita dificuldade em entender como programar, pois sou professora de Matemática e preciso da ajuda de um professor de Informática para me ajudar.";
- P17: "Eu também gostei do que estamos aprendendo e como tenho um colega professor de Informática, que me ajudou, consegui já desenvolver uma atividade para mostrar aos meus alunos!".
- P40: "Na minha escola, só temos dois computadores no laboratório de informática e não funcionam direito!".
- P41: "Eu também não tenho acesso a muitos computadores no meu dia a dia, até porque o nosso laboratório de informática da escola está ruim".

³⁰ Estes relatos foram gravados em áudio e depois transcritos.

- P43: "Não consegui compreender bem ainda como utilizar o *Scratch* (programação) junto com meus alunos".
- P10: "Gostei bastante do *Scratch*, e, se este curso continuar ano que vem, acho que vou fazer de novo, pois tivemos pouco tempo para trabalhar com os alunos neste final de semestre".
- P05: "Eu acho que temos que solicitar para que os nossos professores de informática estejam presentes sempre juntamente com os professores de matemática, para que os mesmos possam ajudar a gente.
- P26: "Estamos no final do ano e tem muita atividade ocorrendo, assim, o tempo para acompanhar o curso está pequeno".

Com base nos relatos da entrevista, foi possível perceber que os professores de Matemática afirmaram precisar da ajuda dos professores de Informática para acompanhar a formação.

Além disso, a questão de tempo disponibilizado para a formação foi questionada pelos professores devido à proximidade do final do ano letivo. O pesquisador esclareceu que a pesquisa teria continuidade no início do próximo semestre.

4.1.3 Terceiro encontro presencial em 02/03/2018 e 16/03/2018

A partir deste encontro, foi efetuada uma nova divisão dos professores participantes em quatro turmas, e cada um dos encontros previstos ocorreu em duas datas com a turma dividida entre os turnos da manhã e da tarde.

Desta forma, buscou-se um atendimento mais individualizado, principalmente para os professores de Matemática que relataram a necessidade de um acompanhamento dos colegas da Informática³¹ demonstrando, assim, maior necessidade de acompanhamento durante a formação.

Como a experiência da utilização do ambiente *GOOGLE Classroom* não foi bem sucedida, a partir deste encontro, todas as atividades propostas passaram a ser efetuadas através do grupo de *Whatsapp*. O Roteiro do terceiro encontro foi dimensionado para 4 horas de duração e dividido nos seguintes momentos:

- Recepção dos professores;
- Retomada do *Scratch*;

³¹ Esta solicitação dos professores de Matemática não pôde ser atendida devido aos horários disponíveis dos professores de Informática e dos próprios professores da Matemática. A solicitação por parte dos professores da Matemática era que estivessem em uma mesma turma que os professores de Informática que atuavam nas mesmas escolas.

- Diálogo sobre as propostas de implementação com o *Scratch*;
- Comandos do *Scratch* que poderiam ser utilizados para implementar as propostas;
- Encerramento.

Devido ao período de recesso dos professores, compreendendo as férias de final de ano, foi necessário uma retomada dos conceitos associados ao *Scratch* e, então, foi proposta a criação em conjunto de um programa simples entre o grupo de professores e o pesquisador.

Como este encontro foi dividido em duas datas nos turnos da manhã e da tarde, o mesmo foi realizado quatro vezes, com quatro turmas distintas. E para cada um destes momentos, foi produzido um programa inicial no *Scratch* para discutir conceitos e comandos.

Foram utilizados comandos de movimento, aparência, eventos, controle e operadores para o desenvolvimento dos programas. Também foram modificadas as propriedades relativas às fantasias do(s) ator(es), como cor e troca de personagens.

A dinâmica do encontro consistia em apresentar um programa desenvolvido pelo pesquisador para os professores, e, posteriormente, os professores utilizavam o exemplo apresentado para desenvolver suas próprias criações. Os professores podiam trabalhar individualmente, em duplas ou em grupos maiores.

Os programas criados pelo pesquisador, que serviram como exemplos para desenvolvimento das atividades junto com cada uma das quatro turmas, foram descritos a seguir:

- 02/03/2018 - turno - manhã Proposta do programa criado: movimentar o ator na tela utilizando o *mouse*; se o mesmo estivesse localizado em uma coordenada x maior que zero, seria exibida uma cor para o ator; se não, a cor mudaria. O programa desenvolvido foi disponibilizado em: <https://scratch.mit.edu/projects/241176885/>.
- 02/03/2018 - turno - tarde Proposta do programa criado: um ator se movimentaria aleatoriamente pelo palco; toda vez que o usuário clicasse sobre o mesmo com o *mouse*, a cor do ator mudaria. O programa desenvolvido foi disponibilizado em: <https://scratch.mit.edu/projects/241761522/>.
- 16/03/2018 - turno - manhã Proposta do programa criado: dois atores se movimentariam aleatoriamente pelo palco; toda vez que os mesmos se tocassem, uma mensagem seria exibida por um dos atores. O programa desenvolvido foi disponibilizado em: <https://scratch.mit.edu/projects/241767947/>.

- 16/03/2018 - turno - tarde Proposta do programa criado: movimentar um ator na tela utilizando o *mouse*; o ator iria indicando qual a soma da sua posição x com a sua posição y. O programa desenvolvido foi disponibilizado em: <https://scratch.mit.edu/projects/244082699/>.

Cada programa criado pelo pesquisador serviu como ponto de partida para o desenvolvimento dos programas dos professores em cada um dos encontros. Com os programas apresentados pelo pesquisador, foi possível demonstrar como utilizar os comandos do *Scratch* para mediar os professores na concepção de suas próprias ideias de programas.

Assim, os professores criavam uma proposta nova de programa ou reutilizavam o que havia sido apresentado pelo pesquisador.

A mediação do pesquisador com os professores buscou estimular a habilidade dos mesmos em aprender, habilidade esta que permite lidar com um desafio ou problema inesperado, assim como na visão construcionista de Papert (2008).

A habilidade de aprender e principalmente de conseguir lidar com um novo desafio ou problema tornou-se fundamental em sala de aula e no âmbito profissional independentemente da área de atuação.

Depois da criação dos programas e desenvolvimento de atividades para trabalhar os conceitos associados a cada comando do *Scratch*, um diálogo sobre as dimensões do Pensamento Computacional foi conduzido com o grupo³², associando, por exemplo, a dimensão do Raciocínio Lógico com a utilização de blocos condicionais “SE” e “SE-SENÃO”.

Com o objetivo de manter a continuidade da formação no período compreendido entre o terceiro e o quarto encontro presencial, e também enriquecer a base de conhecimento dos professores com relação ao *Scratch*, o pesquisador criou vídeos tutoriais com exemplos que exploraram comandos do *Scratch* associando-os com as dimensões do Pensamento Computacional.

A escolha por utilizar vídeos veio após a leitura de um artigo da *Nova Escola* (2009) sobre formação de professores, no qual uma coordenadora pedagógica citava que utilizar a gravação em vídeo possuía maior potencial formativo, pois permite que a prática seja analisada como realmente acontece.

Uma descrição sobre os vídeos desenvolvidos pelo pesquisador foram apresentadas a seguir:

- Utilização da estrutura condicional “Se-Então-Senão”: traz a explicação sobre a estrutura condicional, utilização do evento “bandeira verde”, laço de repetição sempre,

³² Conforme referência da Tabela 1 apresentada no capítulo 2

sensores de pergunta e resposta, comparador “maior que” e comando de aparência “diga”.

O contexto proposto foi a entrada de um número, com ação do ator de informar se o número dado era maior ou menor que 10. O vídeo³³ foi disponibilizado em: <https://www.youtube.com/watch?v=r04BvgcNemM&t=3s>.

Neste vídeo, foram contempladas as seguintes dimensões do Pensamento Computacional: Pensamento Lógico: níveis básico³⁴, desenvolvimento e proficiente; Interatividade do Usuário: níveis básico e desenvolvimento; Controle de Fluxo: níveis básico e desenvolvimento.

- Utilização de estruturas de repetição: traz a explicação sobre as estruturas de repetição “repita determinado número de vezes” e “repita até que”. A estrutura “repita até que” foi utilizada com o contexto de movimentar o ator na tela até o momento em que o mesmo atingisse uma coordenada menor que -100 no eixo x.

O vídeo³⁵ foi disponibilizado em: <https://www.youtube.com/watch?v=Y2p6CUytDZ0&t=56s>. Neste vídeo, foram contempladas as seguintes dimensões do Pensamento Computacional: Pensamento Lógico: nível proficiente; Controle de Fluxo: níveis básico e desenvolvimento.

- Utilização de variáveis: traz a explicação sobre como criar e utilizar variáveis. O contexto proposto foi a criação de um programa para efetuar a soma de dois números inteiros informados pelo usuário.

O vídeo³⁶ foi disponibilizado em: <https://www.youtube.com/watch?v=TgT4UOLJpps&t=105s>. Foram contempladas neste vídeo as seguintes dimensões do Pensamento Computacional: Pensamento Lógico: nível proficiente; Interatividade do Usuário: níveis básico e desenvolvimento; Controle de Fluxo: níveis básico e desenvolvimento; Representação de Dados: níveis básico e desenvolvimento.

Como atividade de entrega foi solicitado aos professores um programa para demonstrar como estava sua compreensão dos comandos discutidos. O tema do programa foi livre, não sendo necessário utilizar comandos específicos, ou qualquer forma de roteiro. Com a data de entrega combinada para 02 de abril de 2018, a ideia foi deixar a sugestão por conta da criatividade dos professores, os programas entregues foram listados no Apêndice D.

³³ Publicado em 19/03/2018.

³⁴ A classificação nos níveis “básico”, “desenvolvimento” ou “proficiente” foram obtidas utilizando o *Dr. Scratch* para classificar os programas explicados em cada vídeo.

³⁵ Publicado em 26/03/2018.

³⁶ Publicado em 27/04/2018.

4.1.4 Quarto encontro presencial em 04/05/2018 e 11/05/2018

Neste encontro, a proposta inicial foi discutir sobre os programas³⁷ que foram entregues (ou não) pelos professores, quais eram as dificuldades em implementá-los, quais foram os comandos utilizados do *Scratch*, de que forma chegaram nas suas propostas de programas e se havia um alinhamento do programa criado com algum conteúdo de sala de aula em específico.

Os professores que não haviam entregue os programas, por algum motivo, puderam desenvolvê-los em sala de aula com o auxílio do pesquisador e dos colegas. A figura 14 mostra o grupo de professores da turma da manhã desenvolvendo suas atividades.



Figura 14 – Turma do turno da manhã desenvolvendo atividades.

Fonte: elaborada pelo autor.

4.1.4.1 Entrevistas

Em um segundo momento, realizou-se uma entrevista semiestruturada com um roteiro que consistiu em um conjunto de perguntas feitas aos participantes, com a interação entre o entrevistador (pesquisador) e os participantes, sendo gravada em áudio para posterior transcrição que foi utilizada para gerar os dados para uma análise de conteúdo. As perguntas utilizadas na entrevista foram:

- Você considera que participar desta formação vai trazer benefícios para sua atividade docente?
- Você considera importante que os seus alunos tenham acesso a uma formação similar a esta?
- Você pensa em desenvolver atividades trabalhadas nesta formação com seus alunos? Em caso negativo, por que não? Em caso positivo, o que você trabalharia?

³⁷ Uma análise dos programas criados será conduzida na seção 4.1.7 apresentada após os relatos dos encontros presenciais.

- Para quem tem intenção de desenvolver atividades do Pensamento Computacional em sala de aula, como você poderia encaixar os tópicos trabalhados nesta formação em sua própria disciplina? Com relação ao tempo, uso laboratório, etc.
- Como você avalia o seu desempenho nesta formação?
- Fale sobre as dificuldades de desenvolvimento de atividades para trabalhar o Pensamento Computacional em sala de aula com seus alunos. Podem ser citados itens como: infraestrutura, apoio institucional, formação suficiente/insuficiente do professor, tempo disponível no cronograma escolar, dificuldade de avaliação de trabalhos de Pensamento Computacional, interesse dos alunos nestas atividades, interesse do próprio professor nestas atividades, currículo, dentre outros.

A condução da entrevista buscou evitar que os participantes utilizassem repostas “óbvias” para as perguntas. Por exemplo: para a pergunta “*Você considera que participar desta formação vai trazer benefícios para sua atividade docente?*”, caso o entrevistado respondesse simplesmente com um “sim”, o pesquisador buscava ampliar a resposta questionando quais eram estes benefícios, se o entrevistado podia dar um exemplo ou ainda se já estava verificando na prática estes benefícios.

O número total de professores que respondeu ao questionário da entrevista desta etapa foi de 32, com 11 professores de Informática (92% do total de professores de Informática) e 21 professores de Matemática (57% do total de professores de Matemática).

4.1.4.2 Análise de Conteúdo

A Análise de Conteúdo é uma técnica sistemática e replicável que consiste em compactar muitas palavras detectadas em um texto, com uma quantidade menor de categorias, utilizando regras explícitas de codificação, permitindo que pesquisadores possam analisar grandes volumes de dados com relativa facilidade e de maneira sistemática (STEMLER, 2001).

Para analisar as falas dos professores, adotou-se a análise de conteúdo com processo de codificação proposta por Saldana (2009), empregando os trechos de texto extraídos das entrevistas como dados. Após a análise dos trechos de textos, foram criados códigos que capturaram a ideia principal do texto analisado. Posteriormente, os códigos foram contabilizados e permitiram analisar de maneira sistematizada as informações.

Os códigos apresentados no quadro 1 foram criados por dois codificadores, separadamente, sendo um codificador, o autor desta tese, e o segundo codificador, um colega de grupo de pesquisa do programa de doutorado.

Quadro 1 – Categorias e Códigos.

Categoria	Código	Descrição
DIRETRIZES	GRADE	Grade curricular da escola e/ou currículo implementado.
	CNTDS	Para cumprir os conteúdos obrigatórios, não há tempo disponível para trabalhar o Scratch.
	LEGIS	Falta documentação/legislação que oriente a implementação da informática nas escolas.
HABILIDADES	HABAL	Habilidade dos alunos no contexto da informática: facilidade de programar, lógica, etc.
	HABIP	Professor referenciou precisar de habilidades no Scratch para aplicar com os alunos.
	DIFIC	Comentários dos professores sobre sua dificuldade com relação à informática.
INTEGRAÇÃO CURRICULAR	INTER	Qualquer referência à interdisciplinaridade.
	IMPLE	Como implementar o Scratch: jogos, projetos, oficina, etc.
RECURSOS DISPONÍVEIS	PRFAX	Necessidade de um profissional que auxilie com questões de informática na escola.
	ADEQO	Infraestrutura da escola está adequada para desenvolvimento de atividades.
	DEFIC	Infraestrutura da escola não está adequada para desenvolvimento de atividades.
	HORAP	Carga horária disponível dos professores para trabalhar com informática.
	HORAL	Horários disponíveis para utilizar os laboratórios de informática da escola.
	NUMAL	Referência ao número de alunos em uma turma.
REAÇÃO DOS PARTICIPANTES	ACETA	Aceitação dos alunos com relação ao Scratch.
	ACETP	Aceitação dos professores com relação ao Scratch.
	REJEA	Rejeição dos alunos com relação ao Scratch.
	REJEP	Rejeição dos professores com relação ao Scratch.
	RESPP	Professor reconhece que deve estudar para compreender como utilizar o Scratch.

Fonte: elaborado pelo autor.

O nível de concordância entre os códigos propostos pelos dois codificadores foi verificado com a ferramenta *online dfreelon*³⁸, que efetuou o cálculo do coeficiente de *Cohen's Kappa*, classificando os níveis de concordância entre codificadores de “Pequeno” (0.00) a “Quase Perfeito” (1.00) (STEMLER, 2001).

A análise dos códigos criados também utilizou uma outra classificação, reunindo-os em categorias, de forma similar ao que foi apresentado nos trabalhos de³⁹ de Bower et al. (2017) e Yadav, Gretter e McLean (2017). Observando que os trabalhos citados utilizaram o *Software NVivo 11* e, nesta tese, o processo foi realizado manualmente pelos dois codificadores.

A primeira análise de concordância dos códigos criados apresentou 28.13% dos valores de coeficiente com classificação acima de “Moderado” (0.41 - 0.60) e buscando melhorar esta classificação foi utilizado um processo de conciliação entre os codificadores.

O processo de conciliação consistiu, basicamente, em uma conversa entre os codificadores para decidir porque um código foi escolhido. E para os códigos que divergiram entre os codificadores, houve uma discussão para verificar o porquê da escolha deste código.

Após discutir sobre os códigos os codificadores, podiam entrar em um consenso e convergir para a escolha de um mesmo código que representasse um trecho de texto com significado para ambos os codificadores.

Este processo de conciliação descrito seguiu de maneira similar à proposta metodológica utilizada por Garrison et al. (2006). Após realizar o processo de conciliação a totalidade da concordância entre os códigos criados pelos codificadores, atingiu classificação igual ou superior à “Significativa”, validando, assim, o processo entre os codificadores.

A figura 15 mostra a quantidade de ocorrências de cada código, com o número correspondente à quantidade de professores que teve a fala classificada.

O número de ocorrências de cada código apresentado na figura 15 trouxe uma série de informações sobre as falas dos professores, assim como as 20 ocorrências do código ACETP (70% relativos as falas dos professores de Matemática), que indicaram boa aceitação da formação e do uso do *Scratch*.

As 19 ocorrências do código PRFAX (79% relativos as falas dos professores de Matemática) foram um indício de que os professores de Matemática não estavam dispostos a se apropriar dos conhecimentos disponibilizados na formação, sem “dependerm” do auxílio de um professor de Informática.

A questão dos professores de Matemática demonstrarem uma “dependência” em relação à presença dos professores de Informática pareceu um comportamento caracterís-

³⁸ <http://dfreelon.org/utils/recalfront/recal2/>, acessado em 01/06/2018.

³⁹ Cf. seção 3.1.1.

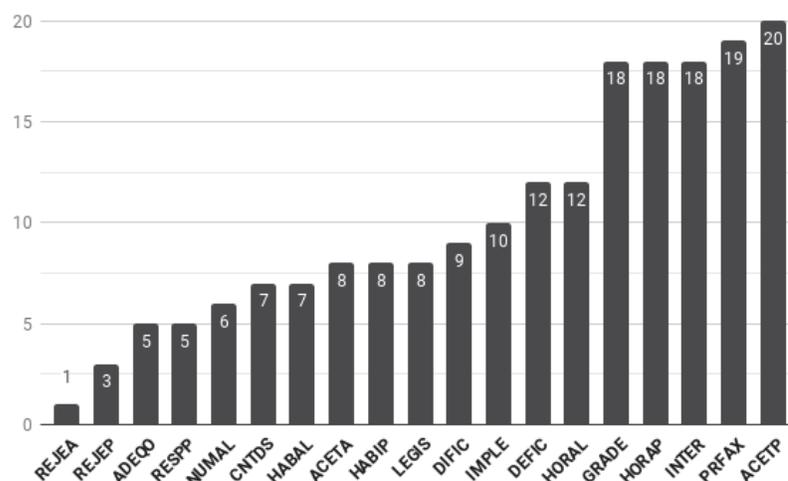


Figura 15 – Quantidade de Ocorrências dos Códigos.

Fonte: elaborada pelo autor.

tico ao realizar-se uma comparação com o que relatou [Oliveira \(2009\)](#) sobre professores menos ambientados com tecnologia apresentarem menor interesse e disponibilidade em aprender sozinhos.

Ainda, o comportamento dos professores de Matemática poderia estar relacionado com a forma como os professores de Informática atuavam nas escolas do município em que a pesquisa foi realizada. A Secretaria de Educação daquela cidade, por sua vez, distribuiu os professores de Informática nas escolas para auxiliar outros docentes das demais disciplinas.

Desta forma, os professores de Informática assumiram um papel de responsáveis por qualquer item relacionado à Informática, e assim se apropriar dos conceitos do Pensamento Computacional poderia ter sido interpretado como uma responsabilidade exclusiva daqueles educadores.

A possibilidade de os professores de Matemática terem avaliado que sua participação na formação de Pensamento Computacional dependeria dos professores da Informática, conforme exposto, foi identificada analogamente por [Leite et al. \(2017\)](#), que trouxeram um relato sobre a demanda das escolas municipais de Francisco Beltrão, no Paraná, onde foi imputado somente aos professores oriundos da Licenciatura em Informática a responsabilidade de trabalhar assuntos relativos ao Pensamento Computacional.

Para verificar se problemas de infraestrutura poderiam estar afetando a aceitação dos professores com relação à formação em Pensamento Computacional e ao *Scratch*, relacionou-se o número de ocorrências do código REJEP (associado com duas ocorrências nas falas dos professores da Matemática e uma dos professores da Informática) com o do código DEFIC. Sendo encontrado somente uma ocorrência de fala que pontuou em ambos

os códigos (professora de matemática P47: “Desmotivada pois falta laboratório”).

Esta única ocorrência de fala sobre problemas de infraestrutura revelou que este item ou estava bem resolvido entre os professores, ou não era um fator tão relevante para os mesmos.

Realizando-se a análise das ocorrências dos códigos ADEQO e DEFIC, que também possuíam relação com infraestrutura, classificou-se que as estruturas das escolas, conforme detectado nas falas dos professores, possuíam um nível adequado, com este termo utilizado para referir-se a uma escola que apresentasse uma estrutura “Adequada” de forma similar à definição de Neto et al. (2013, p. 90):

[...] as escolas deste nível, em geral, possuem uma infraestrutura mais completa, o que permite um ambiente mais propício para o ensino e aprendizagem. Essas escolas possuem, por exemplo, espaços como sala de professores, biblioteca, laboratório de informática [...] Além disso, são escolas que possuem equipamentos complementares como copiadora e acesso à internet

Porém, mesmo com a infraestrutura das escolas estando adequada em um consenso geral, foi constatada resistência dos professores com relação à formação, e este fato permitiu uma analogia com o que foi apresentado por Ma, Andersson e Streith (2005) quando citaram que somente a disponibilidade de infraestrutura de informática não era um item suficiente para garantir que o computador fosse adotado como ferramenta de uso diário dos professores.

Contextualizando a constatação dos autores (MA; ANDERSSON; STREITH, 2005) em relação a esta tese, mesmo com os professores (na grande maioria) possuindo acesso a uma infraestrutura básica para realizarem as atividades do Pensamento Computacional/*Scratch*, este fato não resultou em uma aceitação dos mesmos com relação a se apropriar dos conhecimentos da formação.

Similarmente, Leite et al. (2017) relataram que, mesmo com a disponibilidade de *tablets*, *smartphones* e acesso à *Internet*, professores de escolas pesquisadas demonstraram resistência ao não introduzirem estes elementos em sua prática docente, prevalecendo a cultura do “quadro e giz” dos currículos tradicionalistas.

Os autores reforçaram a necessidade da formação continuada para suprir professores capazes de utilizar a tecnologia disponível e mudar o paradigma dos currículos tradicionais.

Uma análise adicional que permitiu constatar sobre a resistência dos professores de Matemática, durante a formação, consistiu em relacionar o número de professores de cada disciplina (Matemática ou Informática) com o número de entregas de projetos desenvolvidos no *Scratch*, conforme apresentado na tabela 4.

Tabela 4 – Número de professores em relação à entrega de projetos do *Scratch*.

	Professores	%	Número de projetos Scratch entregues	
Matemática	37	75.5	26	70.27% dos professores de Matemática
Informática	12	24.5	12	100% dos professores de Informática
Total	49	100	38	

Fonte: elaborada pelo autor.

Foi possível observar, segundo apresentado na tabela 4, que dos quarenta e nove professores que participaram da formação (com trinta e sete professores da Matemática e doze da Informática), vinte e seis professores da Matemática (correspondendo a 70.27% do total da Matemática) entregaram seus projetos em comparação aos doze professores de Informática (100% do total da Informática).

Dentre os vinte e seis professoras da Matemática, ocorreram oito falas classificadas no código DIFIC, que teve um total de nove ocorrências, ou seja, 88.89% do total de nove ocorrências do código DIFIC foram geradas por professores da Matemática, relatando dificuldades em relação aos conhecimentos tratados na formação.

A constatação sobre resistência por parte dos professores, no contexto da Informática, assim como no caso da formação em Pensamento Computacional relatada nesta tese, pode ser encontrada em trabalhos desenvolvidos há mais de uma década, assim como relatado por [Penteado et al. \(2000, p. 11\)](#):

O contato com professores através de pesquisas e cursos desenvolvidos [...] mostra que grande parte deles ainda conhece pouco de informática e não se sente em condições de incorporar as Tecnologias Informáticas em sua prática. Alguns tendem a encarar com desconfiança e resistência a introdução das novas tecnologias de informação e comunicação.

Uma observação fundamental, realizada em relação aos professores da Matemática, foi que, mesmo com uma porcentagem menor de entregas em relação ao número total de professores do seu grupo, 26 entregas de projetos foi um número expressivo.

Ou seja, mesmo que tenham exposto que tiveram dificuldades em relação aos conhecimentos tratados na formação, conseguiram criar o projeto no *Scratch*, e talvez tenham menos dificuldade do que necessidade em afirmá-la. Similarmente ao que apresentou [Curzon et al. \(2009\)](#) durante o desenvolvimento de atividades de computação desplugada, relatando como comportamento comum dos professores apresentar apreensão quando precisam lidar com um tópico do qual não dominam.

Da mesma forma, conforme relatado nesta tese, a dificuldade dos professores na apropriação de conhecimentos relativos ao Pensamento Computacional caracterizou-se

como um problema de dimensões globais, trazendo como exemplo o trabalho de Almeida e Valente (2019), que apresentaram uma iniciativa do governo da Estônia, onde os professores não se apropriaram das TDIC da forma esperada após uma formação inicial para professores, ficando aquém das expectativas governamentais.

Especificamente com relação à resistência ao *Scratch*, e intrinsecamente à formação em Pensamento Computacional por parte dos professores de Matemática apresentada nesta tese, considerou-se esta como uma situação problemática, pois, no texto da BNCC (2019) das 9 ocorrências do termo “Pensamento Computacional”, 4 delas⁴⁰ foram referenciadas dentro da seção “4.2. A ÁREA DE MATEMÁTICA”.

Isto é, esperava-se⁴¹ uma relação direta da Matemática com o Pensamento Computacional, assim como a relação da Matemática com o contexto computacional em geral, da mesma forma que foi exposto por Morais, Basso e Fagundes (2017, p. 469):

Logo, é fundamental que o professor de matemática vislumbre a possibilidade de promover a aprendizagem de matemática aos seus estudantes quando lhes ensina a programar [...] Não se tem dúvidas que os professores percebem tal fato quando o projeto envolve unicamente conceitos de matemática, mas o que é relevante que ele tome consciência de que ela também é potencializada mesmo quando o conteúdo não é matemático [...] diante do crescente desejo de inserir a ciência da computação na escola básica, pensa-se que o educador matemático pode ser o profissional responsável por tal introdução, [...]

Para reforçar a importância de relacionar a Matemática com o contexto computacional, Brackmann (2017, p. 163-164) trouxe que:

[...] deve ocorrer uma formação de professores para que atuem como multiplicadores do saber [...] países, como os Estados Unidos da América e a Alemanha, encontraram a solução quanto à falta de professores para lecionar a disciplina de Computação nas aulas de Matemática, podendo esse exemplo servir de modelo ao Brasil.

Os autores (MORAIS; BASSO; FAGUNDES, 2017; BRACKMANN, 2017) trouxeram a ideia do professor de Matemática como uma peça fundamental no contexto da utilização dos computadores, e assim seria possível imaginar que estes docentes também pudessem assumir um papel fundamental na condução de atividades do Pensamento Computacional. Entretanto, a realidade encontrada com o grupo de professores da Matemática, que foi pesquisado nesta tese, mostrou uma realidade um pouco diferente.

A percepção da resistência por parte dos professores da Matemática em se apropriar dos conhecimentos referentes ao Pensamento Computacional não ficou evidente so-

⁴⁰ Cf. Capítulo 2.

⁴¹ Observa-se também que a própria discussão sobre a “responsabilidade” em disseminar o Pensamento Computacional é controversa, sendo imputada aos professores de Matemática na BNCC e aos professores licenciados em Informática nas diretrizes Curriculares Nacionais (LEITE et al., 2017).

mente nas observações do pesquisador, mas também pôde ser detectada na própria relação entre os professores durante o curso de formação, assim como explicitado no relato da professora de Informática P37, quando comentou sua relação com a colega (professora de Matemática) de escola: “Faço um trabalho interdisciplinar mas não estou tendo apoio da professora de Matemática, me vejo pouco motivada no sentido da interdisciplinaridade, pois a professora de Matemática da escola não participa ativamente.”

Além do mais, uma vez que o código INTER foi o terceiro em número de ocorrências (junto com os códigos GRADE e HORAP), foi possível perceber que a interdisciplinaridade foi importante na percepção do grupo de professores. Mas situações como a relatada pela professora P37, sobre a falta de interesse da colega de escola, dificultaram o desenvolvimento de atividades relacionadas à formação em Pensamento Computacional.

Com relação ao conceito de interdisciplinaridade considerou-se nesta tese que o mesmo é definido em termos de “atitude de ousadia e busca diante do conhecimento” assim como apresentado por [Fazenda \(1979, p. 149\)](#):

Se definirmos interdisciplinaridade como junção de disciplinas, caberá pensar currículo apenas na formatação de sua grade. Contudo, se definirmos interdisciplinaridade como atitude de ousadia e busca diante do conhecimento, caberá pensar aspectos que envolvem a cultura do lugar onde se formam professores.

Outra percepção sobre a forma como os professores de Matemática reagiram à formação pôde ser obtida relacionando-se o código DIFIC (9 ocorrências) com ao código RESPP (4 ocorrências). Ou seja, o número de professores que reconheceram ter dificuldade durante a formação foi superior ao número de professores que assumiram precisar comprometer-se para superar sua dificuldade.

Indo mais a fundo nesta análise, houve somente uma ocorrência coincidente dos códigos DIFIC e RESPP (na fala da professora P04), cuja ocorrência de RESPP para P04 se relacionou com a sua percepção de ter que: “Contextualizar a Informática na vida dos alunos” e não com relação a se dedicar para sanar suas dificuldades. Ao contrário do que foi citado por P08: “Tenho que estudar nesta área (*Scratch*/Informática)” e P30: “É mais a minha parte agora de estudar [sic]”.

Também mereceram destaque as 8 ocorrências de aceitação dos alunos com relação aos conceitos do Pensamento Computacional/*Scratch* (código: ACETA, 62.5% professores de Matemática) em relação a somente uma ocorrência de rejeição (código: REJEA) ao *Scratch*. Com o caso de rejeição sendo específico onde o aluno não possuía acesso ao computador em casa, corroborando que a utilização da tecnologia computacional depende, antes de tudo, do acesso ao computador no local de trabalho/estudo e em casa ([MA; ANDERSSON; STREITH, 2005](#)).

Ainda, as ocorrências do código ACETA foram indícios de que os professores que

utilizaram o *Scratch* em aula com os alunos obtiveram resultados positivos, conforme retorno dos próprios estudantes. Exemplificando com os relatos de P00: “Ficam encantados (os alunos) pois tem condições de programar” e P10: “Eles (alunos) se interessam muito”, de forma similar a Miotto e Cardoso (2014) que ensinaram o conceito de função e posteriormente relataram o interesse dos alunos, concluindo que o *Scratch* foi um aliado importante no processo de ensino e aprendizagem da Matemática.

Reforça-se também que os professores reconheceram nos alunos a capacidade de apropriar-se da tecnologia, conforme ocorrências do código HABAL. Exemplificando com as falas de P01: “Vão acabar ensinando o professor” e P48: “Eles (alunos) aprendem com muito mais facilidade que a gente (professores)”.

As ocorrências do código IMPLÉ demonstraram que, mesmo insistindo na dificuldade que possuem em Informática, os professores - sejam da Informática, sejam da Matemática - que buscaram utilizar o *Scratch*, conseguiram, de alguma forma, implementar atividades com os seus alunos em sala de aula.

As implementações ocorreram no formato de oficinas, mostra de trabalhos ou projetos, assim como exemplificado pela fala da professora P21: “a gente está desenvolvendo como trabalho com os alunos na forma de um projeto [sic] [...] não teria como trabalhar com todos então a gente deu uma selecionada [...] já no segundo encontro já surpreenderam como eles têm esta habilidade de tentar mexer no computador [sic]”.

Por fim, a ocorrência dos códigos NUMAL, CNTDS, LEGIS, HORAL, GRADE e HORAP mostraram a necessidade da promoção de ambiente favorável à implementação do Pensamento Computacional nas escolas. Estes códigos pertencentes às categorias DIRETRIZES e RECURSOS DISPONÍVEIS estavam diretamente relacionados com as falas dos professores em relação aos itens que dependiam da decisão de níveis hierárquicos superiores (direção das escolas e secretaria de educação).

Sendo necessário, então, que pessoas dos níveis hierárquicos superiores revisassem a forma como conteúdos, carga horária dos professores e disponibilidade de infraestrutura estariam organizados para promover a implementação de oficinas e projetos na temática do Pensamento Computacional. Não só com professores de Matemática e Informática, mas de todas as áreas.

4.1.4.3 Considerações

A análise de conteúdo, juntamente com os códigos que foram criados, permitiram avaliar as impressões dos professores com relação à formação proposta utilizando a ferramenta *Scratch*.

Observou-se que existiu resistência dos professores, principalmente do grupo da Matemática, em aceitar o Pensamento Computacional como uma ferramenta de auxílio

em sala de aula. Sendo este resultado similar ao que foi apresentado por [Penteado et al. \(2000\)](#), com a constatação sobre a resistência à Informática em um trabalho publicado há 20 anos, mostrando que o processo de aceitação da tecnologia é lento em relação ao avanço da mesma.

Além disso, a constatação sobre a resistência à tecnologia, por parte dos professores, é problemática, uma vez que o apoio dos mesmos é fundamental para que se possa disseminar os conceitos do Pensamento Computacional juntos aos alunos.

Verificou-se, também, que a resistência à apropriação dos conceitos do Pensamento Computacional e utilização do *Scratch*, na forma como foi apresentada nesta tese, foi um relato comum encontrado na literatura ([Revista Educação, 2018](#)). A formação insuficiente dos professores na área tecnológica, mais especificamente com relação à apropriação dos conceitos do Pensamento Computacional, impede de haver exploração das potencialidades pedagógicas de novas tecnologias, fazendo com que métodos mais tradicionais sigam sendo reproduzidos.

Foi possível observar que o resultado das entrevistas realizadas deixou evidente, em vários momentos, que as respostas dos professores mesclaram opiniões técnicas sobre o curso de formação, com algumas de suas insatisfações profissionais e situações que vivenciaram na sua atividade docente - o que foi relatado similarmente por [Souza \(2006, p. 486\)](#) em um estudo de caso sobre formação continuada de professores:

Praticamente todas as professoras tomaram a entrevista como uma oportunidade de protestar, desabafar e compartilhar um pouco sobre sua história profissional. Deste modo, ao falarem sobre seu trabalho e sobre o curso que estavam frequentando, as falas dessas professoras era, por vezes, errático, com alterações frequentes de assunto. Manifestações de insatisfação com as condições de trabalho oferecidas nas escolas apresentavam-se em vários graus e de diversas maneiras no grupo. Quando questionadas sobre o curso, várias professoras comentaram a respeito de suas relações com colegas e superiores, problemas com alunos, incertezas e inseguranças ligadas à carreira docente.

Ao final do quarto encontro presencial, uma atividade foi proposta aos professores, consistindo em alterar um programa que foi criado pelo pesquisador, modificando uma ou mais características associadas aos atores utilizados no programa. As características a modificar poderiam ser, por exemplo, a cor ou a fantasia de cada ator.

O programa criado pelo pesquisador⁴² utilizou 2 atores, cada um com seu *script*, trabalhando, desta maneira, com o conceito de paralelismo e com 5 eventos associados, possuindo, então, complexidade superior aos programas trabalhados anteriormente na formação.

⁴² Disponibilizado em: <https://scratch.mit.edu/projects/220590755/>.

O objetivo da atividade proposta foi instigar os professores a desenvolver seus próprios programas com mais elementos e maior complexidade, utilizando comandos do *Scratch* pertencentes às dimensões⁴³ do paralelismo e representação de dados. Os programas entregues pelos professores referentes a esta atividade foram analisados na seção 4.1.7.

4.1.5 Quinto encontro presencial em 06/07/2018 e 13/07/2018

Este foi o último encontro presencial com teor técnico, cuja ideia central foi desenvolver programas contextualizados, conforme necessidades de sala de aula, apresentadas pelos professores.

Previamente ao encontro, foi solicitado, através do grupo de *Whatsapp*, que os professores trouxessem ideias de programas para desenvolver ou programas já prontos que precisassem de alguma modificação.

Durante o encontro, o grupo de professores discutia uma determinada ideia e se a implementação da mesma já havia obtido êxito. Para os casos em que as ideias não haviam ainda sido implementadas, o professor, ou grupo de professores, fazia a exposição da ideia e quais as suas dúvidas com relação à execução.

O grupo contribuía com novas ideias e o conhecimento necessário para a realizá-las com mediação do pesquisador.

Neste encontro, foi observado que os professores da Matemática participaram mais ativamente do desenvolvimento das tarefas, concluindo-se que, devido ao fato de a turma ter sido reduzida em número de participantes, e o pesquisador ter conseguido atender cada professor individualmente, os professores da Matemática ficaram em uma situação mais “confortável”, já que ter o acompanhamento do pesquisador associou-se à forma como o grupo da Matemática costumou agir em parceria com o grupo da Informática.

Porém, nos encontros realizados previamente, o desempenho dos professores da Matemática não havia sido tão ativo, principalmente quando os mesmos eram solicitados a desempenhar tarefas individuais, nas quais não podiam contar com a “assessoria” dos professores da Informática ou do pesquisador.

Para explicar este comportamento, retomamos que a estrutura proposta no município onde a pesquisa foi realizada previa professores de Informática que atendiam uma ou mais escolas, ficando responsáveis pelos laboratórios e condução de atividades no contexto das TICs.

E os demais professores, incluindo àqueles de Matemática, informavam quais atividades estavam desenvolvendo com suas turmas para que os professores da Informática

⁴³ Cf. Moreno-Leon, Robles e Roman-Gonzalez (2015) na seção 2.2.

podessem conduzir atividades com os alunos que contribuíssem com as outras disciplinas.

Por exemplo: as professoras da Matemática podiam solicitar alguma atividade de Informática para reforçar um conteúdo em específico como trabalhar com uma planilha eletrônica para reforçar as operações aritméticas básicas.

O direcionamento qualquer item relacionado à utilização do computador para os professores de Informática, pelos docentes das demais disciplinas, gerou uma sistemática de trabalho que, segundo apontam os professores de Matemática, justifica a resistência em se apropriar dos conhecimentos trabalhados durante a formação em Pensamento Computacional.

Ou seja, os professores de Matemática parecem ter assumido que a dinâmica de trabalho deveria seguir um formato no qual eles seriam responsáveis por trazer a ideia para desenvolvimento de uma atividade, porém, contextualizar o Pensamento Computacional e utilizar o *Scratch* seria uma “responsabilidade” somente dos professores da Informática.

Os programas desenvolvidos neste encontro foram analisados quantitativamente na seção 4.1.7 para verificar estatísticas relacionadas às dimensões do Pensamento Computacional, que foram apresentadas nos programas.

4.1.6 Sexto encontro presencial 22/10/2018

Este foi o encontro de encerramento das atividades onde ocorreu a entrega do certificado da formação em Pensamento Computacional para os professores que obtiveram o desempenho solicitado, pois, conforme estabelecido junto à Secretaria de Educação, a certificação dos docentes ficou condicionada à participação em sala de aula e à entrega de pelo menos um dos três programas que foram solicitados durante a formação.

Os professores que entregaram 1, 2 ou 3 dos programas solicitados receberam respectivamente 75%, 85% e 100% de participação na formação. O modelo do certificado gerado foi apresentado no Apêndice B, contendo os tópicos desenvolvidos e a carga horária correspondente.

O encontro reuniu os professores, profissionais da Secretaria de Educação e demais envolvidos com o pesquisador que palestrou sobre a pesquisa desenvolvida, junto com alguns professores que trouxeram relatos de sua participação na formação.

A figura 16 mostra um momento da apresentação de relatos dos professores, quando estes trouxeram informações sobre as atividades que haviam realizado em suas escolas, com relação ao que aprenderam na formação e quais eram suas impressões sobre a formação em um contexto geral.



Figura 16 – Apresentação de relatos dos professores.

Fonte: elaborada pelo autor.

4.1.7 Análise dos Programas

Nesta seção foram analisados os programas desenvolvidos durante o curso de formação pelos professores, utilizando o *Scratch*. Esta análise foi de caráter quantitativo, utilizando a ferramenta *Dr. Scratch* para pontuar os programas entregues.

Através da análise da pontuação dos programas, foi possível inferir de que forma os professores conseguiram converter os conhecimentos adquiridos na formação de Pensamento Computacional em programas que pudessem ser utilizados em suas atividades de sala de aula.

Os programas avaliados foram desenvolvidos em três momentos distintos durante a formação. O primeiro⁴⁴ programa criado buscou verificar como foi o contato inicial dos professores com o *Scratch*. O segundo⁴⁵ programa permitiu avaliar como os professores exploraram novos comandos em um programa com complexidade superior, desenvolvido pelo pesquisador e modificado pelos professores.

E, por fim, o terceiro⁴⁶ programa buscou verificar de que forma os professores evoluíram com relação aos programas anteriores desenvolvidos e qual foi a capacidade dos professores em propor um programa que tivesse aplicabilidade em sala de aula.

Destacou-se que, nestes três momentos de criação de cada um dos programas propostos, os professores foram conduzidos pelo pesquisador de forma a utilizar exemplos

⁴⁴ Cf. seção 4.1.3.

⁴⁵ Cf. seção 4.1.4.

⁴⁶ Cf. seção 4.1.5.

discutidos em sala de aula e reaproveitar programas que já haviam sido desenvolvidos por eles.

Ou seja, reutilizando, por exemplo, trechos de programa criados e ideias já desenvolvidas por colegas, pelo próprio professor ou pelo pesquisador, seguindo a filosofia de “remixar” e de compartilhar (*share*) o que os *Scratchers* criam e disponibilizam na comunidade *online*.

Todos os programas que foram produzidos, durante a formação pelo pesquisador, professores e pelos alunos, foram compartilhados na nuvem, podendo ser acessados por usuários do *Scratch*, constituindo, assim, um material de referência para os professores trabalharem com seus alunos futuramente, atendendo o que foi recomendado por Meerbaum-Salant, Armoni e Ben-Ari (2010), que citaram a importância de os professores terem acesso a materiais didáticos de qualidade, que pudessem ajudar os professores no desenvolvimento de conteúdos.

Evitando, assim, sempre ter que começar do zero o desenvolvimento de um novo conteúdo, o que é muito importante para professores que tenham pouca formação em assuntos como CC e Pensamento Computacional. Assim como foi o caso dos sujeitos desta pesquisa, principalmente em relação aos professores da Matemática.

Incentivou-se, ainda durante a criação dos programas, que os mesmos fossem criados no formato de jogos, propiciando uma boa aceitação por parte dos alunos que participariam de atividades com os professores em sala de aula.

Destacando que, ao criar os jogos, os professores tiveram a oportunidade de testar novos comandos no *Scratch*, mesclando diferentes conhecimentos e promovendo a interdisciplinaridade Matemática \times Informática.

Sobre a criação de atividades no formato de jogos, Almeida e Valente (2019, p. 216) destacaram que:

A criação de jogos digitais ou *games* pode ser vista como uma atividade rica para a aprendizagem, com o potencial de permitir a integração de diferentes áreas do conhecimento como: a estética, [...], uso de som, música, cores; [...] e a tecnologia, os *softwares* usados bem como os dispositivos que executam o *game*.

No Apêndice D foi apresentada a lista dos 3 programas que foram entregues por cada um dos professores, cuja previsão total de entregas era de 147 programas (3 programas \times 49 professores) mas, conforme pode ser observado pelos itens “sem entrega” da lista, foram entregues 115 programas.

Mais especificamente, dos 49 professores que participaram da formação, somente a professora de Matemática P29 não realizou a entrega de pelo menos um programa, o

que era uma condição necessária para receber a certificação da formação⁴⁷.

Porém, mesmo após a professora ter citado que gostou da formação, ela disse que já estava aposentada e que a certificação não seria de seu interesse. Sendo assim, considerou-se que esta foi uma decisão pessoal da professora associada provavelmente ao seu planejamento de carreira.

Este fato foi comparado com o trecho do texto de Miranda (2012, p. 19): “O sentido do desenvolvimento dos professores depende das suas vidas pessoais e profissionais [...]”.

Uma vez que nem todos os professores entregaram os três programas solicitados, foi efetuada uma filtragem na lista de programas do Apêndice D. Nesta filtragem, foram removidos os professores que não entregaram o primeiro ou o terceiro programa.

O objetivo pós-filtragem foi observar a diferença de pontuação do primeiro para o terceiro programa e verificar se houve, ou não, uma evolução dos professores com relação aos conceitos do Pensamento Computacional avaliados com o *Dr. Scratch*. A filtragem realizada foi similar a proposta de Bower et al. (2017)⁴⁸, que consideraram em sua pesquisa somente dados dos participantes que responderam os questionários em momentos pré e pós *workshops* realizados.

Ainda, com relação à lista de programas apresentada no Apêndice D, o segundo programa não foi considerado já que o mesmo^{49,50} havia sido criado pelo pesquisador para que os professores realizassem modificações no mesmo e, sendo assim, a pontuação atribuída a este programa no *Dr. Scratch* teria influencia direta do pesquisador.

A lista de programas resultante foi apresentada na tabela 5, onde as 13 células da coluna “Código do Professor” destacadas com texto em negrito se referem aos professores de Informática, e as demais 23 células, aos professores de Matemática.

O histograma apresentado na figura 17 trouxe uma análise dos dados disponibilizados na tabela 5, com a pontuação total obtida da soma dos dois programas entregues por cada professor.

Uma vez que a distribuição de frequência dos dados representados no histograma foi assimétrica, utilizou-se a análise da Mediana⁵¹ com valor quinze e evidenciou a diferença da pontuação mínima (nove pontos) dos dois professores da Matemática em relação à pontuação máxima (vinte e sete pontos) de um professor de Informática.

A análise da mediana confirmou a expectativa de que os professores de Informática obtivessem uma pontuação maior, já que possuíam maior vivência com conceitos similares aos do Pensamento Computacional e maior experiência com lógica de programação,

⁴⁷ Cf. seção 4.1.6.

⁴⁸ Cf. seção 3.1.1.

⁴⁹ Utilizado como exemplo na seção 4.1 (figura 8).

⁵⁰ Cf. seção 4.1.4.

⁵¹ Assim como recomendado por Rodrigues, Lima e Barbosa (2017).

Tabela 5 – Pontuação do *Dr. Scratch* para o primeiro e o terceiro programa.

Código do Professor	Pontos Dr. Scratch Programa 1	Pontos Dr. Scratch Programa 3	Código do Professor	Pontos Dr. Scratch Programa 1	Pontos Dr. Scratch Programa 3
P00	6/21	5/21	P22	9/21	10/21
P01	5/21	16/21	P23	8/21	11/21
P02	8/21	7/21	P24	4/21	5/21
P03	8/21	5/21	P25	8/21	16/21
P05	8/21	5/21	P26	7/21	16/21
P06	4/21	5/21	P27	5/21	9/21
P07	4/21	7/21	P28	9/21	10/21
P10	5/21	5/21	P30	6/21	6/21
P12	6/21	9/21	P31	4/21	11/21
P13	6/21	9/21	P33	6/21	11/21
P14	3/21	8/21	P36	7/21	11/21
P15	5/21	8/21	P37	9/21	7/21
P16	14/21	11/21	P38	9/21	7/21
P17	7/21	12/21	P39	9/21	7/21
P18	7/21	8/21	P40	6/21	9/21
P19	8/21	6/21	P42	11/21	6/21
P20	5/21	5/21	P44	7/21	9/21
P21	9/21	10/21	P49	11/21	14/21

Fonte: elaborada pelo autor.

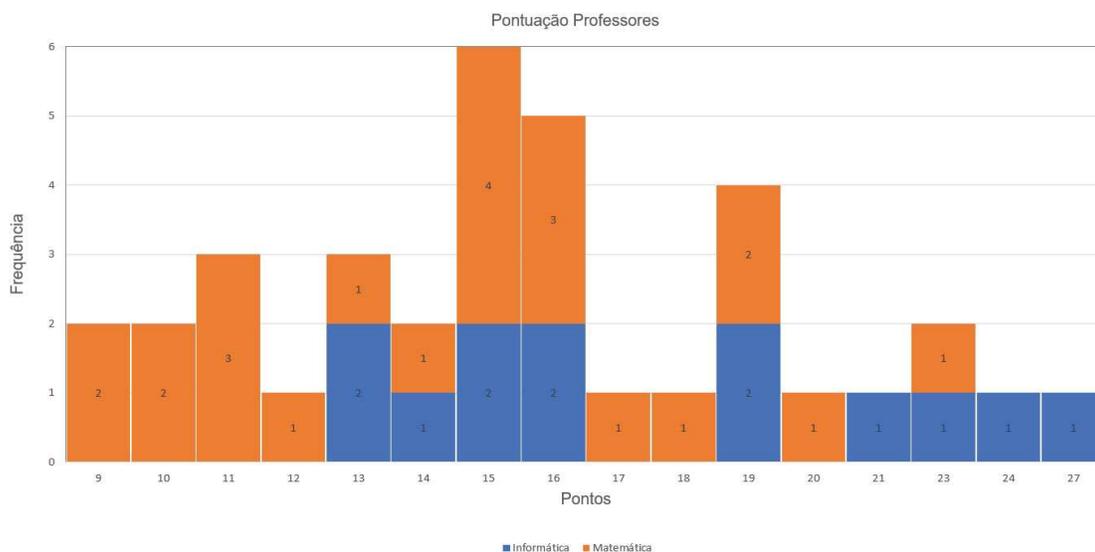


Figura 17 – Pontuação total dos professores para o primeiro e o terceiro programa.

Fonte: elaborada pelo autor.

obtendo, conseqüentemente, melhor desempenho com o *Scratch*.

Além da pontuação mínima e máxima, foi possível observar pontuações significativas para ambas as formações (Matemática e Informática), exemplificando com os vinte e três pontos de um professor de Matemática. Como esta pontuação foi gerada de uma parceria da Matemática com a Informática, serviu para validar a ação conjunta das pro-

fessoras.

Com as professoras de Matemática trazendo o contexto para o programa, e as professoras da Informática auxiliando no desenvolvimento do código, com as professoras de ambas as formações trabalhando interdisciplinarmente o Pensamento Computacional.

A relação do Pensamento Computacional \times programação, encontrada na parceria da Matemática com a Informática, foi similar à descrição de [Digital Promise \(2017, p. 19\)](#): “A habilidade necessária para informar ao computador o que fazer é programar. O processo de pensamento por trás da programação é o pensamento computacional⁵²”.

E uma vez que o foco principal da formação era que os professores trabalhassem com os conceitos do Pensamento Computacional, com a programação no *Scratch*, sendo uma ferramenta complementar de implementação, a parceria da Matemática com a Informática atendeu perfeitamente o objetivo.

Uma análise da pontuação dos programas (referenciados na tabela 5) em relação às dimensões do Pensamento Computacional⁵³ está apresentada⁵⁴ na tabela 6.

Tabela 6 – Pontuação média das dimensões do Pensamento Computacional.

	Programa 1		Programa 3		P3 - P1	
	Matemát.	Informát.	Matemát.	Informát.	Matemát.	Informát.
Lógica	1.17	1.77	0.87	1.38	- 0.30	- 0.38
Paralelismo	0.00	0.23	0.39	0.85	0.39	0.62
Interatividade	1.52	1.85	1.96	1.92	0.43	0.08
Representação de dados	1.26	1.92	1.91	1.69	0.65	- 0.23
Controle de fluxo	1.70	1.85	2.04	1.92	0.35	0.08
Sincronização	0.43	0.69	0.78	1.00	0.35	0.31
Abstração	0.09	0.31	0.39	0.69	0.30	0.38

Fonte: elaborada pelo autor.

Ao calcular a média geral de pontuação do Programa 1 e do Programa 3, obteve-se, respectivamente, 7,03 e 8,75. Valores estes que demonstraram um aumento na pontuação geral dos professores, corroborando uma influência positiva da formação.

Os números que foram apresentados após a utilização do *Dr. Scratch* serviram como uma métrica auxiliar para identificar como os professores conseguiram implementar suas ideias no *Scratch*, onde, mesmo para os professores que obtiveram uma pontuação menor, não houve um significado relativo de que o programa não tenha servido, conforme objetivos definidos.

⁵² Minha tradução para o texto original: *The skill required to tell a computer what to do is programming. The thought process behind programming is computational thinking.*

⁵³ Apresentadas na tabela 1 do Capítulo 2.

⁵⁴ A pontuação detalhada de cada item foi apresentada no apêndice E.

Pois, assim como afirmaram Romero, Lepage e Lille (2017), a ferramenta *Dr. Scratch* é capaz de fornecer boas dicas para melhorar a qualidade de um programa avaliado, identificando, por exemplo, os nomes de elementos que não foram adicionados e blocos repetidos. Mas não consegue “medir” o quanto de criatividade há no programa e nem se o mesmo é funcional.

No caso dos autores citados⁵⁵, a medida de criatividade adotada⁵⁶ em sua pesquisa foi verificar se blocos não apresentados aos participantes haviam sido utilizados em seus programas.

Como o *Dr. Scratch* não mede criatividade, mesmo com uma pontuação relativa menor, o programa criado por um professor pode ter utilidade para o mesmo em sala de aula, como uma ferramenta para sua prática didática ou para fixação de algum conceito estudado na formação.

A análise com o *Dr. Scratch* gerou dados que permitiram avaliar quantitativamente como os professores, de cada uma das formações (Matemática ou a Informática), conseguiram utilizar os conceitos do Pensamento Computacional no *Scratch*.

Destaca-se que uma análise qualitativa dos programas criados pelos professores na etapa (posterior) de visitaç o nas escolas foi realizada e apresentada na se o 4.2 desta tese.

Analisando separadamente o desempenho dos professores com forma o em Matem tica, e com forma o em Inform tica, e comparando o aumento na pontua o do primeiro para o terceiro programa, obteve-se, respectivamente, as m dias de 35,2% e 9,8%, demonstrando que os professores das duas forma es tiveram um aumento relativo em seu desempenho.

O fato de os professores de Inform tica possuirem maior desenvoltura com programa o, j  no in cio da forma o, explica o menor aumento relativo da pontua o. Dito isto, podemos destacar que os professores de Matem tica, mesmo apresentando resist ncia   forma o, conseguiram aumentar significativamente sua pontua o.

Ainda, o aumento percentual inferior dos professores de Inform tica foi devido ao fato que, desde o primeiro programa, j  obtiveram pontua es maiores, mantendo uma pontua o similar no terceiro programa e, conseqentemente, tendo uma varia o menor.

Com rela o  s pontua es obtidas para as sete dimens es do Pensamento Computacional, conforme apresentado na tabela 6, verificou-se que, com exce o da dimens o da L gica, todas as demais dimens es tiveram um aumento relativo.

A explica o para que a dimens o da L gica n o tenha obtido um aumento significativo foi que, tanto no primeiro, quanto no terceiro programa, os professores utilizaram

⁵⁵ Ibidem.

⁵⁶ Cf. se o 3.1.1.

extensivamente a estrutura SE-ENTÃO-SENÃO, a qual pontuou na dimensão da Lógica no *Dr. Scratch*.

Um dado que conseguiu demonstrar que os professores deram maior ênfase à compreensão da estrutura SE-ENTÃO-SENÃO foi o número de acessos aos dois vídeos disponibilizados no *YouTube* antes da data de entrega do programa ⁵⁷. O vídeo que trazia a explicação sobre a estrutura SE-ENTÃO-SENÃO teve 311 visualizações e o vídeo sobre laço de repetição, 125 visualizações.

Conforme observado nas entregas do programa três, os professores demonstraram que já dominavam outros conceitos do Pensamento Computacional e, conseqüentemente, conseguiam utilizar novos comandos do *Scratch* associados às outras dimensões além da Lógica, como foi o caso do Controle de Fluxo, que obteve a maior pontuação para o programa 3.

Concluiu-se que, inicialmente, a dimensão da Lógica foi a mais utilizada sendo posteriormente mesclada com outras dimensões, o que resultou em uma redução relativa de sua utilização e um aumento correspondente das demais dimensões.

Uma vez que, durante o desenvolvimento dos programas avaliados, a interação entre os professores das duas formações foi constante, buscou-se saber qual a relação dos resultados de cada uma das formações em relação às dimensões do Pensamento Computacional.

Para isto, foi utilizado o coeficiente de correlação de *Pearson* para interpretar se haveria uma relação linear, indicando que uma variação da pontuação de um conceito, para os professores da Matemática, estaria associada diretamente com a variação na pontuação para os professores da Informática.

Calculando o coeficiente de *Pearson* para os valores médios da diferença da pontuação de cada conceito do Pensamento Computacional, apresentados na tabela 6, obteve-se ⁵⁸ $R = 0.3676$. Este valor indicou uma correlação fraca (FILHO; JÚNIOR, 2009) que, dentro do contexto apresentado, pode ser associado ao fato de que os programas criados pelos professores das duas formações, assim como já era esperado, possuíam estruturas diferentes. Com interpretações diferentes e, conseqüentemente, utilizando diferentes conceitos do Pensamento Computacional e blocos de programação correspondentes.

Analisando diretamente os dados, foi possível compreender esta correlação fraca observando, por exemplo, os valores com maior variação positiva, que, para o grupo da Matemática, foi de 0,65 no conceito Representação de dados, e para a Informática, 0,62 no conceito Paralelismo.

⁵⁷ Cf. seção 4.1.3

⁵⁸ Utilizando a calculadora *online* disponibilizada em http://www.gyplan.com.br/pt/correlation_pt.html.

Antes desta análise quantitativa da pontuação dos programas, tomando como referência somente a fala dos professores, e analisando suas ações em sala de aula, uma conclusão sobre o período de formação era de que os professores da Matemática, na grande maioria, haviam tido uma participação pequena na produção dos programas.

Porém, cruzando as pontuações relativas de cada conceito do Pensamento Computacional com as formações (Matemática ou Informática), foi possível verificar que a influência da Matemática nos programas criados foi significativa.

Tomando como exemplo o aumento percentual da Representação de Dados, para o qual os professores de Matemática tiveram dificuldade de compreensão durante a formação, esperava-se uma pontuação maior para os professores de Informática.

Porém, com a pontuação maior tendo sido da Matemática, percebeu-se que suas ideias de programas (principalmente os que realizavam cálculos) utilizaram variáveis, que são blocos do *Scratch* pontuados na dimensão da Representação de Dados.

A utilização das variáveis conduziu, então, a uma situação na qual o conceito de Representação de Dados fosse utilizado e compreendido pelos professores de Matemática, mesmo nos casos onde a implementação dos programas foi realizada com o auxílio dos colegas da Informática.

Ainda, conforme resultados analisados, concluiu-se que, qualitativamente, os professores de Matemática apresentaram uma visão mais conservadora do que os professores de Informática sobre a formação. Já que enfatizaram as dificuldades que encontraram, seja com relação aos conceitos discutidos, seja com relação à implementação destes conceitos utilizando o *Scratch*.

Foi perceptível a relação de dependência dos professores de Matemática em relação aos professores de Informática, já que o grupo da Matemática afirmava necessitar de auxílio da Informática para desenvolver os programas solicitados em várias ocasiões durante a formação.

A análise quantitativa realizada em relação aos programas permitiu observar que o desempenho dos professores de Matemática foi similar aos da Informática em termos absolutos de pontuação.

Confirmando que a resistência, apresentada na fala dos professores de Matemática sobre ter dificuldade com a utilização da ferramenta *Scratch*, não correspondeu aos resultados avaliados, pois os professores de Matemática conseguiram desenvolver as atividades solicitadas e pontuar em todos os conceitos do Pensamento Computacional.

Percebeu-se que a resistência demonstrada pelos professores de Matemática em relação à formação aparentou ser um fator emocional e não técnico, uma vez que, na avaliação quantitativa, os mesmos apresentaram resultados similares aos dos professores

da Informática.

Assim como a resistência à informática, descrita por⁵⁹ [Penteado et al. \(2000\)](#), constatou-se que, nesta tese, mesmo com resultados positivos na formação, a resistência foi recorrente na fala dos professores da Matemática.

Talvez a resistência dos professores de Matemática tenha sido motivada por seu conservadorismo, de forma similar ao que foi descrito por [Reinaldo et al. \(2016, p. 775\)](#) sobre professores que foram analisados enquanto utilizavam *Smartphones* como ferramenta educacional:

A utilização da informática e da computabilidade pelas TICs na educação está relativamente ociosa, devido aos receios e preconceitos dos educadores. Durante as sessões, alguns professores se sentiram assustados quanto à proposta do uso destes dispositivos “desconhecidos” e foram enfaticamente contra. Muitos educadores insistem em acreditar que qualquer ferramenta que chame a atenção dos alunos possa então substituí-los. [...] Entre os educadores há duas correntes distintas: os conservadores e os inovadores. Os conservadores sentem-se pressionados e impõem sempre algum tipo de resistência às mudanças.

Uma comparação com o trabalho de [Romero, Lepage e Lille \(2017\)](#) demonstrou que a formação proposta nesta tese foi mais ampla em termos de materiais apresentados e atividades desenvolvidas com os participantes, tanto quanto ao Pensamento Computacional, como na utilização do *Scratch*, e gerou, conseqüentemente, programas no *Scratch* com pontuações superiores.

Nos programas avaliados pelos autores, citou-se pontuações máximas de dez pontos para somente dois participantes e pontuação de seis pontos para noventa e um participantes.

Nos resultados do grupo apresentado nesta tese, ocorreu uma pontuação mínima de 3, máxima de 14 e média de 7,03 pontos para o primeiro programa. E pontuação mínima de 5, máxima de 15 e média de 8,75 pontos para o terceiro programa entregue.

Destaca-se que esta comparação não foi, de forma alguma, uma comparação “direta” com o trabalho de [Romero, Lepage e Lille \(2017\)](#), uma vez que os objetivos dos trabalhos, sujeitos da pesquisa e demais condições eram bem distintas.

Mas, uma vez que o ambiente de programação e ferramenta de pontuação foram as mesmas utilizadas nesta tese, a comparação buscou evidenciar a influência de cada formação nas pontuações apresentadas pelos participantes.

⁵⁹ Cf. seção 4.1.4.2.

4.2 Visitação e Atividades nas Escolas

Nesta seção foram descritas as atividades realizadas nas escolas onde os professores, participantes da formação em Pensamento Computacional, atuavam. Seis escolas foram visitadas, conforme indicação realizada pela Secretaria de Educação, e mais uma visita foi realizada a convite de um professor de Informática.

Durante estas visitas foram coletados dados que permitiram avaliar como os professores utilizaram os conhecimentos adquiridos na etapa de formação. Os dados coletados foram gerados por meio de entrevistas e análise de programas criados no *Scratch*.

A formatação desta etapa pode ser comparada com a etapa de implementação nas escolas propostas⁶⁰ por Eloy et al. (2017), e também com a situação idealizada por Curzon et al. (2014)⁶¹ de avaliar, posteriormente, a forma como os professores aproveitaram o que aprenderam.

A visitação nas escolas permitiu vivenciar o dia a dia dos professores, proporcionando aos mesmos aplicar os conhecimentos da etapa de formação em sala de aula junto aos seus alunos.

Assim como proposto por Gabini e Diniz (2012, p. 346): “[...] é importante destacar que o desenvolvimento de atividades durante a ação de formação continuada, [...] é um fator determinante para resultados positivos desse processo [...]”.

A escolha das escolas visitadas foi realizada pela Secretaria de Educação que emitiu um *e-mail* informando sobre a visita do pesquisador aos diretores das escolas selecionadas.

O acerto de datas entre as escolas e o pesquisador foi realizado diretamente pelas partes. A Tabela 7 traz as datas em que ocorreram as visitas em cada uma das escolas, com as datas previstas e as datas em que as visitas/atividades efetivamente ocorreram.

Tabela 7 – Datas de visitas e atividades realizadas nas escolas.

Visita na escola	Data prevista	Data realizada	Escola
22/04/2019	15/05/2019	não realizado	GH
24/04/2019	08/05/2019	08 e 15/05/2019	BJ
06/05/2019	18/06/2019	17/09/2019	JR
13/05/2019	23 e 30/05/2019	23 e 30/05/2019	DL
22/05/2019	12 e 15/08/2019	12 e 15/08/2019	ST
31/05/2019	04 e 11/07/2019	04 e 11/07/2019	LS
—	—	09/12/2019	EK

Fonte: elaborada pelo autor.

Observam-se diferenças entre datas previstas e de realização devido aos ajustes que foram solicitados pelas escolas. A seguir, foram relatadas as visitas a cada uma das escolas e quais as atividades que foram realizadas nas mesmas.

⁶⁰ Cf. seção 3.1.1.

⁶¹ Cf. seção 3.1.

4.2.1 Escola GH

A primeira visita na escola GH ocorreu em 22 de abril de 2019, quando o pesquisador foi atendido pelas professoras P25 (Informática) e P26 (Matemática), sem a participação da direção ou da orientação pedagógica. A escola dispunha de um laboratório de Informática com 25 computadores e acesso à *Internet*.

Com relação às atividades que foram desenvolvidas desde a formação, a professora P25 trabalhou com exercícios do site code.org. Já a professora P26 não desenvolveu atividades devido à incompatibilidade de horários das aulas de Matemática com a disponibilidade de horários no laboratório de Informática.

A questão de incompatibilidade de horários, citada por P26, coincidiu com a fala da mesma, que gerou⁶² ocorrência do código HORAL.

Após reunião e diálogo sobre possibilidades de utilização do *Scratch* em sala de aula, a professora P26 sugeriu utilizar um dos programas criados durante a formação em Pensamento Computacional junto com os alunos. O retorno à escola, por parte do pesquisador, foi combinado para o dia 15 de maio de 2019 para acompanhar uma turma do oitavo ano nos períodos de Matemática do turno da tarde. Porém, não foi possível realizar a atividade na data prevista por solicitação das professoras.

Após sucessivos reagendamentos - que também foram cancelados - optou-se pela não realização das atividades previstas para não comprometer o cronograma previsto para a redação da tese.

4.2.2 Escola BJ

A primeira visita na escola ocorreu em 24 de abril de 2019, quando o pesquisador foi atendido pelas professoras P21 (Matemática), P22 (Informática) e pelo vice-diretor. A escola dispunha de um laboratório de Informática com sete computadores parcialmente funcionais com acesso à *Internet*.

A conexão com a *internet* de 2 *MBytes* era compartilhada com a rede administrativa da escola, sendo inadequada para realizar atividades *online* com o *Scratch*, o que causou diversos eventos de instabilidade de conexão e atraso na produção dos programas durante realização das atividades.

Nesta escola, houve por parte das professoras a iniciativa de desenvolver atividades relacionadas ao Pensamento Computacional, desde 2008, enquanto estavam cursando a formação, conforme mencionado na seção 4.1.4.2.

O formato de atividade utilizado pelas professoras foi o de uma oficina para explorar conceitos do Pensamento Computacional e demonstrar a utilização do *Scratch* com

⁶² Cf. seção 4.1.4.2.

um grupo de dez alunos, em horários que eram destinados para a realização de aulas de reforço de Matemática.

A utilização dos horários do reforço de Matemática foi uma alternativa encontrada já que as professoras não conseguiram a disponibilização de outros horários para realizar estas atividades.

Após conversar com as professoras e o vice-diretor, houve o interesse em uma atuação do pesquisador junto com as professoras e alunos que participavam da oficina.

Com o acompanhamento do pesquisador, tendo o objetivo de coletar dados e poder contribuir com as atividades já desenvolvidas na escola. O retorno do pesquisador foi combinado para os dias 08 de maio de 2019 e 15 de maio de 2019, com as atividades realizadas nas datas previstas.

Foi fundamental citar que o *deficit* de recursos materiais, apresentado na escola, foi compensado pela atuação das professoras que organizaram e conseguiram manter a oficina de Pensamento Computacional utilizando os conhecimentos adquiridos na formação e conduzindo atividades com o grupo de alunos.

O primeiro item que mereceu destaque na atuação das professoras foi uma iniciativa de produzir chaveiros (figura 18) com uma referência ao *Scratch* e distribuí-los entre os participantes da oficina.

Os chaveiros serviram para destacar os alunos que participavam do projeto e geravam curiosidade para outros alunos que poderiam, posteriormente, demonstrar interesse em também participar das atividades.

Esta ação das professoras de distribuir os chaveiros foi totalmente alinhada com a ideia de utilizar os “*Badges*” ou emblemas⁶³, que funcionam como uma ferramenta motivacional para os alunos que participam de determinada aula ou projeto (Magna Publications, 2014).

A utilização dos *Badges* foi citada por Basogain et al. (2017), que descreveram um ambiente virtual de aprendizagem que permitia implantação dos *Badges*.

A professora de Informática criou uma sistemática de trabalho em que desenvolveu um exemplo no *Scratch* com determinado número de comandos de lógica e estrutura condicional SE-ENTÃO-SENÃO e, posteriormente, utilizou o exemplo para instigar os alunos para criarem propostas similares.

⁶³ O termo *Badge* pode ser traduzido como crachá, distintivo e emblema dentre outros. Com muitas comunidades *online* utilizando os *Badges* para destacar participantes que atingem determinado patamar ou contribuem de forma significativa. A Fundação Mozilla, desenvolvedora do *web browser Firefox*, por exemplo cita o *Badge* digital: “registro *online* [...] monitorado por uma comunidade em que o beneficiário tenha interagido e obtido o emblema, bem como o trabalho feito para obtê-lo”. <https://bit.ly/mozbadge>, acessado em 27/06/2019.



Figura 18 – Chaveiro produzido na escola BJ para os participantes da oficina.

Fonte: elaborada pelo autor.

A figura 19a mostra a tela inicial do programa⁶⁴ que consistiu em um jogo de solução de labirintos onde o ator deveria se deslocar até um ponto em que terminava a fase atual e avançava para a próxima fase. No percurso realizado pelo ator, no labirinto, era possível capturar “ratinhos” que incrementavam a pontuação do jogador.

A proposta do jogo de solução de labirintos trouxe ludicidade e os alunos foram expostos ao conceito de Lógica, enquanto isso, no programa, se fazia um teste condicional para verificar se o ator estava tocando as paredes do labirinto.

Esta lógica foi implementada com um sensor de cor que detectava quando o ator tocava na cor preta. Finalizando a lógica, se o ator estivesse tocando na cor preta, e, conseqüentemente, nas paredes do labirinto, aquele percurso era reiniciado.

A questão da ludicidade foi positiva assim como exposto⁶⁵ por Fuste e Schmandt (2019, p. 19), quando afirmaram que⁶⁶ “A experimentação lúdica é essencial para prosperar com sucesso no processo criativo”.

E considerando que, na escola BJ, os alunos apresentaram a maior média de pontuação dentre todas as escolas visitadas, permitindo inferir que a forma como o programa foi contextualizada, junto aos estudantes, influenciou os resultados apresentados.

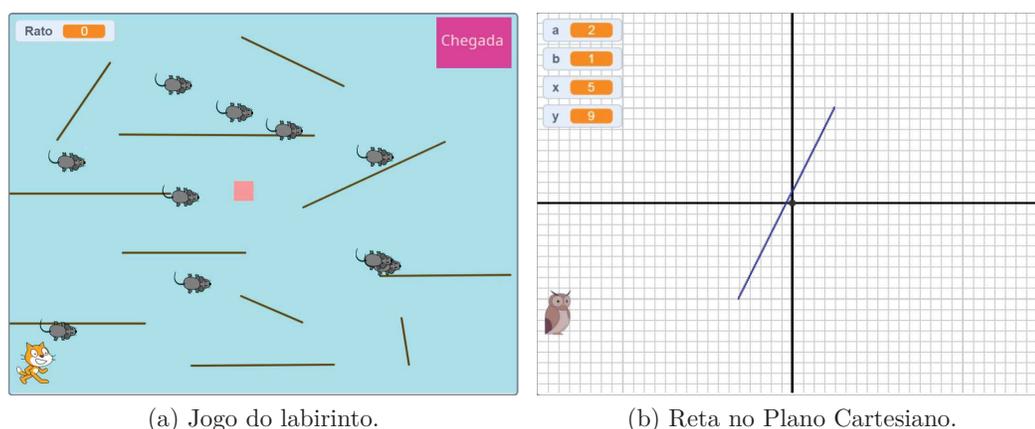
O programa criado pela professora foi contextualizado junto aos alunos que propuseram outros programas com formato similar, posteriormente. Durante a participação do pesquisador nas oficinas, foi possível auxiliar os alunos e professoras resolvendo algumas dúvidas que surgiram na criação dos programas, como, por exemplo, na utilização da estrutura condicional SE-ENTÃO-SENÃO para ativar o início de uma nova fase em um jogo.

Após a proposta inicial da professora de Informática, através da qual os alunos tiveram o primeiro contato com o *Scratch*, a professora de Matemática pôde conduzir um

⁶⁴ Disponibilizado em <https://scratch.mit.edu/projects/313325552>.

⁶⁵ Cf. seção 2.3.2.

⁶⁶ Minha tradução para o texto original: “*Playful experimentation is key in order to successfully thrive in the creative process.*”.



(a) Jogo do labirinto.

(b) Reta no Plano Cartesiano.

Figura 19 – Projetos *Scratch* das professoras na escola BJ.

Fonte: elaborada pelo autor.

experimento reutilizando os conceitos do Pensamento Computacional que já haviam sido estudados e exercitados.

A atividade proposta na Matemática foi um programa para explorar a plotagem da reta no plano cartesiano⁶⁷, modificando os parâmetros a e b da equação da reta.

A figura 19b mostra a tela do programa⁶⁸ proposto, destacando que este foi um programa “remixado”, ou seja, um programa desenvolvido por outro *Scratcher* e que foi utilizado pela professora de Matemática como base para criar o seu próprio programa.

Destacou-se ainda a ação das professoras para organizar as atividades da oficina, mantendo um diário de bordo dos alunos do projeto. Neste diário, foram armazenadas informações como o *login* e senha para acesso *online* ao *Scratch*, para cada aluno (figura 20a), além de um passo a passo para elucidar dúvidas sobre os programas desenvolvidos (figura 20b).

Nas datas em que o pesquisador atuou na escola, foi possível perceber que a oficina de Pensamento Computacional foi bem organizada, com objetivos definidos e participação ativa dos alunos. As figuras 21a e 21b mostram, respectivamente, a ação do pesquisador e da professora junto aos alunos durante uma das visitas. Três programas que foram produzidos pelos alunos foram analisados e discutidos na seção 4.4.

Os programas produzidos pelas professoras, na oficina, foram avaliados quantitativamente com a pontuação gerada no *Dr. Scratch* e qualitativamente através da verificação do contexto e propósito dos programas, seguindo a proposta⁶⁹ de Brennan e Resnick (2012), que permitiu verificar a intenção da(s) professora(s) ao produzir os programas e

⁶⁷ Atendendo as habilidades EF08MA07 e EF08MA08 propostas pela BNCC (pág. 313).

⁶⁸ Disponibilizado em <https://scratch.mit.edu/projects/307085729>.

⁶⁹ Cf. seção 2.4.

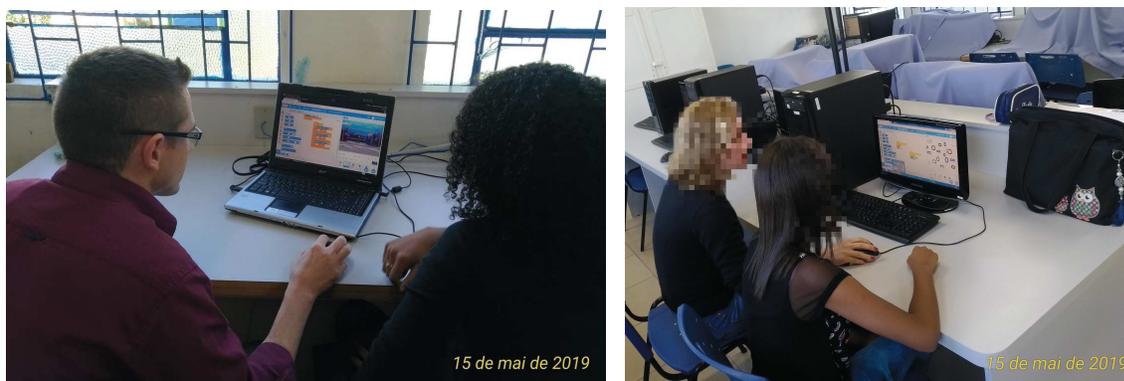


(a) Diário de bordo.

(b) Passo a passo sobre os programas criados.

Figura 20 – Informações dos Projetos criados na escola BJ.

Fonte: elaborada pelo autor.



(a) Pesquisador e aluna.

(b) Professora e aluna.

Figura 21 – Ação do pesquisador e professora com alunas na escola BJ.

Fonte: elaborada pelo autor.

qual o domínio das mesmas com relação aos blocos de programação utilizados no *Scratch*.

Esta análise foi realizada também para todos os demais programas produzidos pelos professores nas outras escolas visitadas.

4.2.3 Escola DL

A primeira visita à escola ocorreu em 13 de maio de 2019, quando o pesquisador foi atendido pelas professoras P03 (Informática), P05 (Matemática), P09 (Matemática) e

pela supervisora educacional.

A escola dispunha de um laboratório de informática com doze computadores funcionais, porém com um *hardware* bem antigo. O acesso à *Internet* apresentava-se nas mesmas condições que na escola BJ, com instabilidade de conexão que causou várias situações de travamento para acesso *online* ao *Scratch*.

As professoras não desenvolveram atividades sobre o Pensamento Computacional previamente à visita do pesquisador, e, nesta escola, diferentemente do que ocorreu nas demais, a professora de Informática não atuou junto com as professoras de Matemática.

Uma vez que as falas de ambas as professoras P05 e P09 geraram ocorrências⁷⁰ do código PRFAX, não foi surpresa que não tivessem implementado atividades sobre o Pensamento Computacional sem o auxílio da professora de Informática, confirmando a resistência em utilizar o *Scratch*.

Assim, após conversar com as professoras e a supervisora educacional, foi proposta a realização de uma oficina piloto de Pensamento Computacional utilizando o *Scratch*, com a participação das professoras de Matemática e de doze alunos que seriam selecionados em turmas dos sétimos, oitavos e nonos anos.

O retorno à escola, por parte do pesquisador, foi combinado para os dias 23 e 30 de maio de 2019, prevendo a realização de uma oficina com 2 horas de duração, em cada uma das datas.

As professoras e a supervisora comentaram a dificuldade de adaptar os horários da grade escolar para este tipo de atividade “extraclasse”. As atividades da oficina tiveram que ser realizadas durante os períodos da disciplina de Matemática⁷¹, justificando a ocorrência⁷² do código GRADE, gerado pela fala da professora P05.

As atividades na escola ocorreram conforme as datas previstas, quando o pesquisador conversou com os alunos sobre os conceitos de abstração e algoritmos e, posteriormente, demonstrou a utilização dos blocos básicos de movimentação, controle e variáveis do *Scratch*.

Foi criado um programa para movimentar um ator com relação aos eixos x e y, trabalhado, assim, com o conceito de Plano Cartesiano⁷³. Também foi adicionado um segundo ator ao programa, gerando, assim, um programa com maior complexidade que permitiu aos alunos trabalhar com a estrutura básica de um jogo.

No jogo que foi proposto, um dos atores podia arremessar um objeto (terceiro

⁷⁰ Cf. seção 4.1.4.2.

⁷¹ Observando que esta situação foi muito similar ao que ocorreu na escola BJ (seção 4.2.2) onde as professoras precisaram utilizar o horário que era destinado ao reforço de Matemática para conduzir a oficina de Pensamento Computacional

⁷² Cf. seção 4.1.4.2.

⁷³ Atendendo habilidades como EF07MA19 proposta pela BNCC (pág. 309).

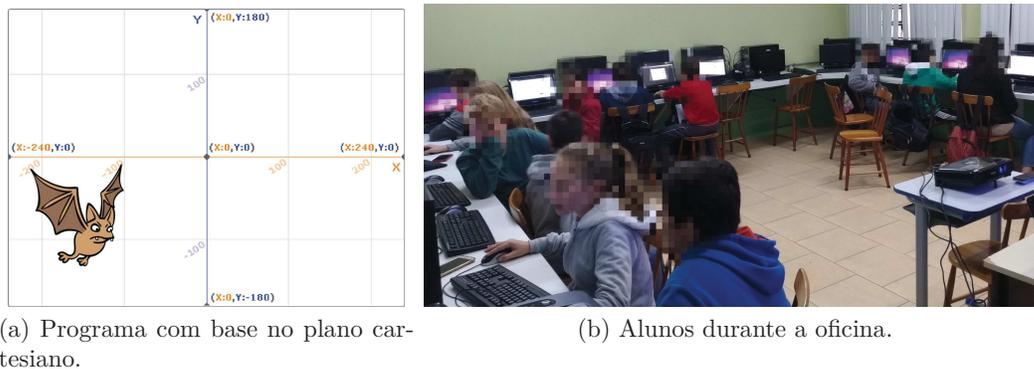


Figura 22 – Ações com os alunos na escola DL.

Fonte: elaborada pelo autor.

ator) no outro e as ocorrências de intersecção de posições, entre atores, eram pontuadas, utilizando, deste modo, o conceito de variáveis e mantendo a premissa da ludicidade, conforme⁷⁴ a proposta desenvolvida na escola BJ. O programa desenvolvido⁷⁵ está representado na figura 22a.

A construção do programa foi realizada pelo pesquisador com as professoras, participando somente como espectadoras, às quais até possuíam as noções básicas de utilização do *Scratch*, mas apresentavam grande dificuldade em sugerir modificações no programa relacionando estas modificações com os blocos de programação.

Para estimular uma participação mais ativa das professoras, o pesquisador utilizava alguns blocos de programação e questionava as professoras e os alunos sobre a forma como o programa iria funcionar.

Após o pesquisador finalizar o programa exemplo, foi solicitado aos alunos que desenvolvessem a sua versão do programa, realizando modificações conforme sua criatividade.

Durante a atividade proposta, o pesquisador e as professoras ficavam auxiliando os alunos com eventuais dúvidas. A figura 22b mostra um dos momentos durante a realização da oficina.

Seis programas produzidos pelos alunos foram analisados na seção 4.4, observando que, mesmo com a participação de 12 alunos na oficina, devido aos problemas enfrentados na conexão com a *internet*, bem como problemas com os computadores, somente seis alunos conseguiram terminar as atividades propostas, dentro do prazo determinado.

⁷⁴ Cf. seção 4.2.2.

⁷⁵ Disponibilizado em <https://scratch.mit.edu/projects/314092995>.

4.2.4 Escola LS

A primeira visita na escola ocorreu em 31 de maio de 2019, quando o pesquisador foi atendido pela professora de Informática P37. A escola dispunha de um laboratório com 14 computadores e conexão com a *Internet*, porém, devido a uma incompatibilidade da versão do *web browser* instalado no sistema operacional *Linux Educacional*⁷⁶, com a versão *online* do *Scratch*, só foi possível utilizar o *Scratch* no modo *offline*.

Com relação às atividades que haviam sido desenvolvidas previamente à visita do pesquisador, a professora P37 utilizou o *Scratch* para ensinar conceitos básicos de lógica e criar atividades no formato de jogos, assim, os alunos já possuíam boas noções de como utilizar os blocos de programação.

Todavia não houve uma utilização interdisciplinar dos conhecimentos junto à professora P39, de Matemática, destacando-se que a mesma optou por não participar da oficina.

A professora P37 já havia sinalizado este comportamento da professora de Matemática previamente⁷⁷ durante a etapa de formação.

Após conversar com a professora P37, foi proposta uma atividade com os alunos dos nonos anos, nos períodos de Informática, e o retorno à escola por parte do pesquisador foi combinado para os dias 04 e 11 de julho de 2019, com as atividades na escola ocorrendo conforme as datas previstas.

Nas datas em que o professor atuou na escola, a professora de Informática trouxe a informação de que o conteúdo que estava sendo desenvolvido na Matemática era referente ao cálculo do Δ da equação do segundo grau⁷⁸, e este assunto foi utilizado para desenvolvimento de um programa exemplo, que permitiu explorar o conceito de variáveis no *Scratch*, conforme registrado na figura 23a, e a figura 23b traz uma imagem do programa⁷⁹ desenvolvido para calcular o Δ .

O desenvolvimento do programa exemplo foi realizado em conjunto pelo pesquisador e professora de Informática.

Depois que o programa exemplo foi apresentado pelo pesquisador, os alunos tiveram liberdade para produzir um jogo, animação ou até mesmo uma modificação com base no programa exemplo que foi desenvolvido pelo pesquisador.

Oito programas criados pelos alunos foram analisados e discutidos na seção 4.4. A figura 24 mostra um registro do pesquisador interagindo com os alunos durante a

⁷⁶ Utilizado em grande parte dos computadores das escolas visitadas.

⁷⁷ Cf. Seção 4.1.4.2.

⁷⁸ Atendendo as habilidades EF08MA09, EF09MA09, EM13MAT302, EM13MAT402, EM13MAT502, EM13MAT302, EM13MAT402 e EM13MAT502 propostas pela BNCC (2019, p. 313 - 543)

⁷⁹ Disponibilizado em <https://scratch.mit.edu/projects/324577038/>.

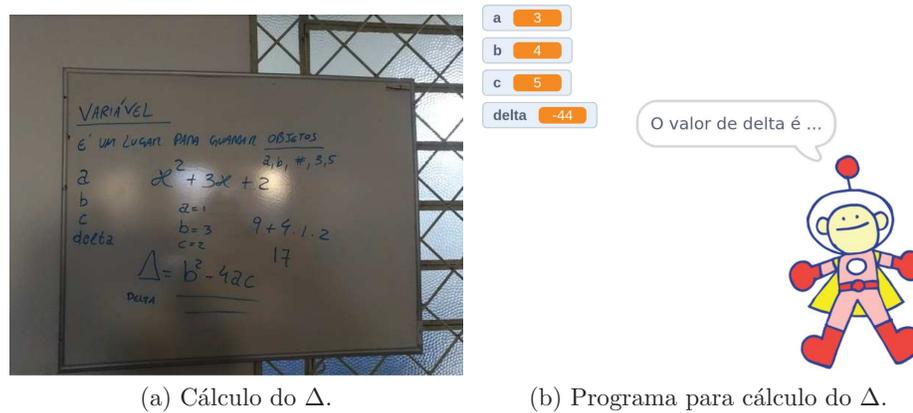


Figura 23 – Ações na escola LS.

Fonte: elaborada pelo autor.

explicação do programa exemplo.



Figura 24 – Interação do pesquisador com os alunos na escola LS.

Fonte: elaborada pelo autor.

4.2.5 Escola ST

A primeira visita na escola ocorreu em 22 de maio de 2019, quando o pesquisador foi atendido pelos professores P00 (Matemática), P06 (Matemática), P23 (Informática) e pela supervisora educacional. A escola dispunha de um laboratório com 23 computadores e acesso à *Internet*.

Com relação às atividades desenvolvidas previamente à visita do pesquisador, o professor de Informática utilizou o *Scratch* para ensinar conceitos básicos de lógica, associando-os à resolução de problemas matemáticos.

Destacou-se que a utilização do *Scratch* foi um dos temas escolhidos durante uma mostra de trabalhos que ocorreu em Julho de 2018. A figura 25 mostra o cartaz referenciando o *Scratch* exibido na mostra de trabalhos.

Após conversar com os professores e com a supervisora educacional, foi proposta uma atividade com alunos dos sextos e sétimos anos nos períodos de Informática e o retorno do pesquisador foi combinado para os dias 12 e 15 de agosto de 2019.



Figura 25 – Mostra de trabalhos na escola ST destacando o *Scratch*.

Fonte: elaborada pelo autor.

As atividades na escola ocorreram conforme as datas previstas e a turma do sétimo ano participou em 12 de agosto de 2019, com a professora P06, que estava desenvolvendo estudos sobre o deslocamento na reta numérica com sua turma.

E a turma do sexto ano participou em 15 de abril de 2019, com a professora P00, que conduzia com a sua turma estudos sobre a potenciação. Em cada uma das datas, o pesquisador utilizou os conhecimentos que estavam sendo trabalhados na Matemática para criar programas exemplos.

Para o estudo sobre deslocamento na reta numérica, foi desenvolvido o programa⁸⁰ representado na figura 26a, onde o usuário do programa informava o número de unidades (passos que o ator iria realizar sobre a reta) desejado e o sinal numérico correspondente, enquanto o ator se deslocava na reta no eixo horizontal.

E para o estudo da potenciação, o programa⁸¹ representado na figura 26b, onde o usuário informava uma base e um expoente obtendo o resultado da exponenciação.

⁸⁰ Disponibilizado em: <https://scratch.mit.edu/projects/324604773>

⁸¹ Disponibilizado em: <https://scratch.mit.edu/projects/324730922>

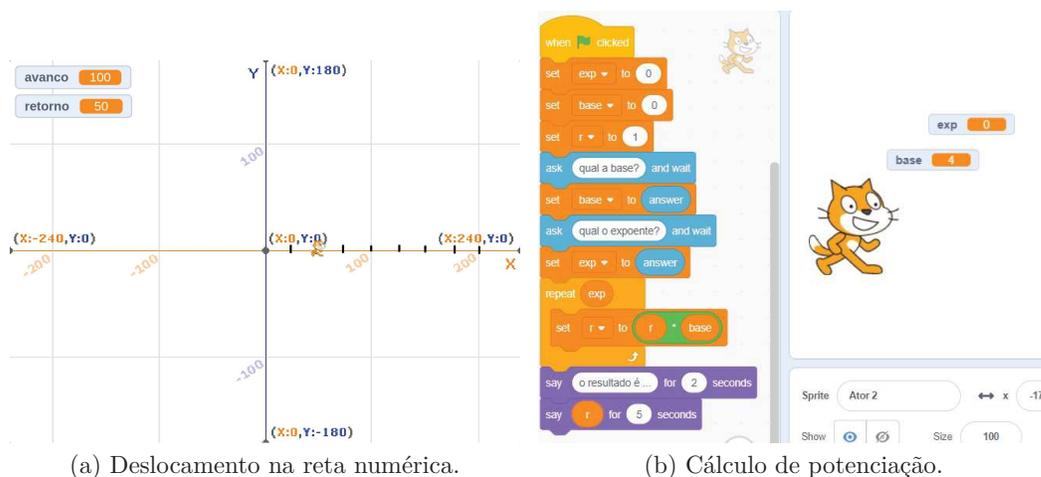


Figura 26 – Programas criados na escola ST.

Fonte: elaborada pelo autor.

Em cada uma das datas que o pesquisador atuou na escola, após a explicação dos programas exemplos, foi solicitado que os alunos criassem suas propostas, e por uma solicitação das professoras de Matemática, os estudantes das duas turmas dedicaram-se a resolver problemas matemáticos similares àqueles que estavam sendo estudados, utilizando o *Scratch*. 18 programas produzidos pelos alunos foram analisados e discutidos na seção 4.4.

Algumas observações sobre a escola precisam ser destacadas. Iniciando pela estrutura, que, comparada àquelas de escolas anteriores, possuía um laboratório com mais computadores funcionais e uma conexão com a *Internet* que se mostrou funcional e possibilitou trabalhar sem travamentos, com o *Scratch*, no modo *online*. O que justificou a ocorrência⁸² do código ADEQO gerado pela fala da professora P00.

Ainda, durante a ação do pesquisador junto aos alunos, as professoras de Matemática agiram somente como espectadoras, similarmente ao que ocorreu⁸³ com as professoras na escola DL.

Analisando as ocorrências⁸⁴ do código PRFAX geradas pelas falas das professoras P00 e P06, foi possível perceber que as mesmas sentiram-se confortáveis em relação ao colega da Informática, entendendo que ele conseguia atender às necessidades do grupo no sentido de que cada professor cuidava de sua área.

Então, por mais que tenha existido um ambiente interdisciplinar, as professoras de Matemática não atuaram com o Pensamento Computacional, apenas indicaram ao professor de Informática quais os conteúdos que ele poderia utilizar com os alunos, e

⁸² Cf. seção 4.1.4.2.

⁸³ Cf. seção 4.2.3.

⁸⁴ Cf. seção 4.1.4.2

assim as atividades desenvolvidas com o *Scratch* foram realizadas somente pelo professor de Informática.

4.2.6 Escola JR

A primeira visita na escola ocorreu em 06 de maio de 2019, quando o pesquisador foi atendido pelas professoras P15 (Informática), P46 (Matemática) e pela supervisora educacional. A escola dispunha de um laboratório de informática com 22 computadores com acesso à *Internet*.

Com relação às atividades que haviam sido desenvolvidas previamente à visita do pesquisador, a professora P15 trabalhou com conceitos de lógica utilizando o *Scratch* e desenvolveu atividades associadas aos conteúdos de Matemática solicitados pela professora P46. Como, por exemplo, a criação de um programa para demonstrar a equação da parábola (equação do segundo grau).

O programa⁸⁵ proposto pelas professoras está representado na figura 27. Após conversar com as professoras e a supervisora educacional, houve interesse em uma atuação do pesquisador em um período de Matemática para desenvolver uma atividade com os alunos do nono ano, com o retorno à escola, por parte do pesquisador, combinado para o dia 18 de junho de 2019.

As atividades na escola ocorreram conforme as datas previstas, quando o pesquisador conduziu uma oficina no formato de “bate-papo” com os alunos que, em sua grande maioria, apresentaram já estar ambientados com a utilização do *Scratch*, demonstrando o bom desempenho do trabalho das professoras.

Nesta escola, assim como ocorreu na escola BJ⁸⁶, observou-se que as professoras trabalharam de forma interdisciplinar e conseguiram desenvolver atividades do Pensamento Computacional com os alunos.

Destacou-se, também, que os alunos tinham boas noções sobre o que era um algoritmo, assim como compreendiam a função de variáveis e possuíam um bom domínio de blocos de lógica, conseguindo utilizar, por exemplo, a estrutura condicional SE-ENTÃO e SE-ENTÃO-SENÃO em diferentes contextos.

A figura 28 mostra dois momentos de interação do pesquisador com os alunos. Seguindo a proposta já realizada nas outras escolas, foi desenvolvido um programa com a dinâmica de um jogo. Com uma atuação constante de interação das professoras e dos alunos, que respondiam prontamente aos questionamentos do pesquisador e praticamente moldaram a proposta do jogo.

⁸⁵ Disponibilizado em <https://scratch.mit.edu/projects/317785071>.

⁸⁶ Cf. seção 4.2.2.

Junto com a criação da proposta do jogo pelo pesquisador, os alunos já começavam a desenvolver sua própria versão do jogo e 16 programas produzidos pelos alunos foram analisados e discutidos na seção 4.4.

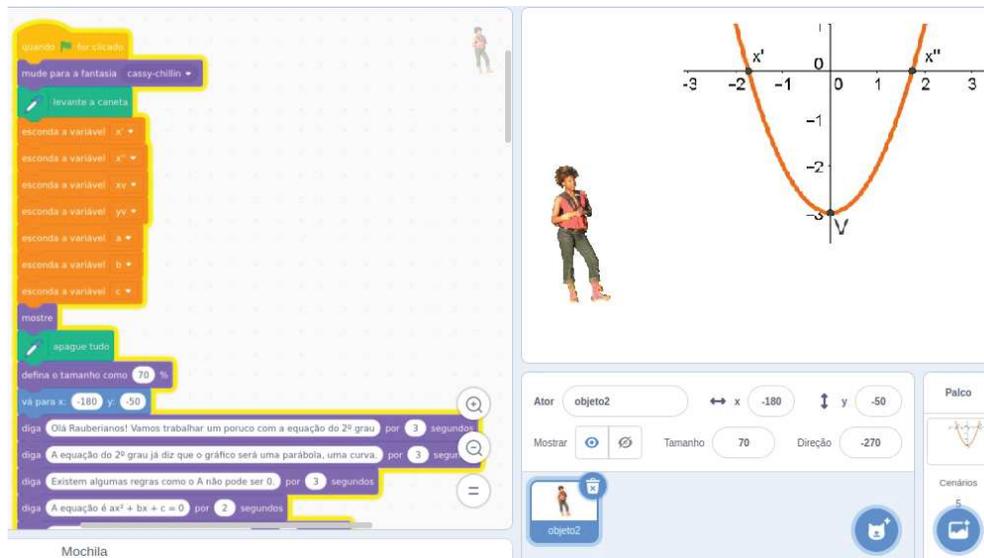


Figura 27 – Programa sobre equação do segundo grau.

Fonte: elaborada pelo autor.



Figura 28 – Interação do pesquisador com os alunos.

Fonte: elaborada pelo autor.

4.2.7 Escola EK

Esta visita não foi prevista pelo pesquisador, sendo realizada em 9 de dezembro de 2019 devido ao convite do professor de Informática P49, que solicitou a participação do pesquisador durante uma mostra de trabalhos. A escola visitada fica localizada em um município vizinho àquele de realização da pesquisa.

Como parte da mostra de trabalhos, alguns alunos criaram propostas com conceitos de computação desplugada, apresentação de vídeos e programas produzidos com o *Scratch*, cujo objetivo da participação do pesquisador na mostra consistiu em avaliar os trabalhos que foram produzidos com o *Scratch*.

A figura 29 mostra dois momentos de interação com os alunos, destacando que a escola possuía uma infraestrutura simples, com poucos equipamentos no laboratório de informática, porém, a forma como o professor P49 conduziu as atividades propiciou a obtenção de trabalhos muito interessantes.

Dentre os trabalhos, destacou-se a participação de alunos PcD⁸⁷, atuando na explicação e demonstração sobre jogos desplugados, além da apresentação de vídeos que demonstraram que os alunos, mesmo com poucos recursos no laboratório, se empenharam para produzi-los.

Considerou-se acrescentar o registro desta visita no texto da tese, pois, assim como na escola BJ, a escola EK foi uma das que possuía menos recursos disponíveis em comparação às demais escolas visitadas, fato que demonstrou a importância da atuação do docente para atingir resultados.

As atitudes dos professores das escolas BJ e EK podem ser exemplos do professor reflexivo caracterizado⁸⁸ por Mello (2000), que inquietaram-se, indignaram-se e mostraram que podiam fazer a diferença junto aos seus alunos.



Figura 29 – Mostra de trabalhos.

Fonte: elaborada pelo autor.

4.3 Análise de Conteúdo

Com o intuito de verificar como os professores reagiram à etapa de visitação nas escolas, foi realizada uma entrevista para coleta de dados com uma posterior análise de conteúdo com o mesmo formato apresentado na seção 4.1.4.2, onde, novamente, a codificação foi realizada por dois codificadores com um posterior processo de conciliação, que resultou na totalidade da codificação, com classificação igual ou superior à “Significativa”.

⁸⁷ Sigla para “Pessoa com Deficiência”, conforme sugerido em http://www.portaldeacessibilidade.rs.gov.br/uploads/1313497232Manual_de_Redacao_AL_Inclusiva.pdf.

⁸⁸ Cf. Introdução do Capítulo 3.

Para gerar os dados para análise de conteúdo da etapa de visitação nas escolas, as seguintes perguntas foram utilizadas durante a entrevista com os professores:

- 1 - Como esta segunda etapa da formação (visita do pesquisador na escola) contribuiu para sua atividade docente?
- 2 - Como você avalia a contribuição desta atividade para os alunos?
- 3 - Você pretende seguir utilizando atividades similares a esta com seus alunos? Em caso negativo, por que não? Em caso positivo, como você trabalharia?
- 4 - Você considera que a atividade desenvolvida conseguiu atender o propósito de trabalhar os conceitos do Pensamento Computacional e também atender os requisitos de sua disciplina?
- 5 - Você considera que conseguiu utilizar os conhecimentos adquiridos na formação de Pensamento Computacional em sala de aula e contribuir para ampliar o conhecimento dos seus alunos?
- 6 - Qual ou quais foram as maiores dificuldades para trabalhar o Pensamento Computacional junto aos seus alunos?

As respostas dos professores foram gravadas em áudio e depois transcritas para serem posteriormente analisadas pelos codificadores. A figura 30 traz a quantidade de códigos apurados, conforme fala dos professores P00, P05, P09, P21, P39 e P46 da Matemática, e P15, P22, P23 e P27 da Informática, que participaram das atividades realizadas durante a visitação nas escolas.

Os códigos foram analisados da mesma forma que aqueles apresentados na seção 4.1.4.2, e assim, para comparar os códigos das duas etapas, a figura traz a quantidade de ocorrências de cada código da etapa metodológica anterior (formação em Pensamento Computacional) e da posterior (visitação nas escolas).

Destaca-se que, durante o processo de codificação e conciliação da etapa de visitação nas escolas, os codificadores trabalharam de forma a utilizar os mesmos códigos e categorias apresentados no quadro 1. Desta forma, foi possível comparar os dois momentos em que o processo de codificação foi aplicado.

Sendo, o primeiro momento⁸⁹, a análise de conteúdo realizada durante o curso de formação e o segundo momento⁹⁰, a análise de conteúdo realizada durante a visitação nas escolas. Como se trata de uma comparação entre os dois momentos, os números de ocorrências dos códigos apresentados na figura 30 são relativos somente aos professores

⁸⁹ Cf. seção 4.1.4.2.

⁹⁰ Cf. seção 4.3.

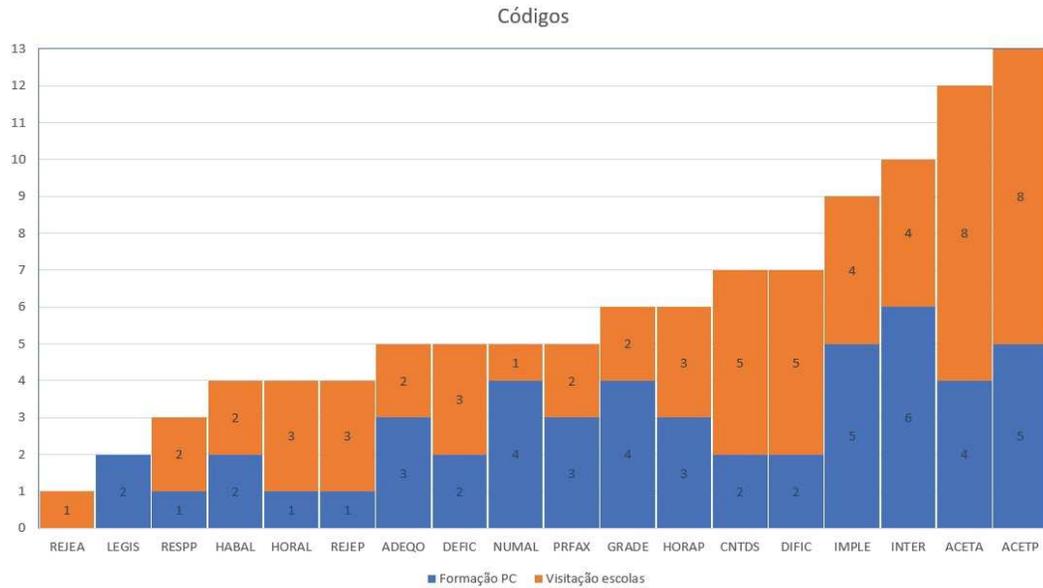


Figura 30 – Quantidade de ocorr ncias dos c digos pr /p s visita o nas escolas.

Fonte: elaborada pelo autor.

que participaram das entrevistas durante o curso de forma o e tamb m posteriormente na etapa de visita o das escolas.

A an lise do n mero de ocorr ncias, de cada c digo, trouxe uma s rie de informa es sobre a fala dos professores, como, por exemplo, os n meros de ocorr ncias dos c digos ACETA e ACETP, os quais tiveram aumentos percentuais⁹¹ respectivos de 100% e 60%, demonstrando que, na etapa de visita o nas escolas, a aceita o dos professores com rela o   forma o e, intrinsecamente,   utiliza o do *Scratch* teve maior destaque em suas falas.

E conforme observa es realizadas pelo pesquisador, durante as atividades com os professores, o que propiciou a maior aceita o dos mesmos foi o fato de que estavam “em seu *habitat*” e vivenciaram resultados positivos junto com seus alunos.

Os professores conseguiram perceber, na pr tica, os benef cios da utiliza o dos conceitos relativos ao Pensamento Computacional, bem como do *Scratch* junto aos alunos, em especial quando relacionaram que o desenvolvimento de programas contribuiu para a compreens o de alguns conceitos matem ticos.

Destacando uma situa o que ocorreu na escola ST, quando foi criado o programa exemplo sobre potencia o, um aluno fez a seguinte observa o: “*Agora eu entendi ... tem que pegar o n mero de baixo e fazer multiplica o quantas vezes tem no n mero de cima*”, onde o “n mero de baixo” era a base e o “n mero de cima”, o expoente.

⁹¹ O aumento ou redu o percentual foi calculado em rela o   ocorr ncia de cada c digo da etapa de visita o em rela o   etapa de forma o.

Assim como pode ser exemplificado no relato⁹² da professora P21 em resposta à quarta questão do questionário: “Muitos, eu percebo muitos alunos conseguindo associar coisas que eu não conseguia atingir eles na Matemática [sic]. E eles conseguiram assimilar a entender melhor o conceito em função do que a gente está fazendo nas nossas oficinas. Eles conseguiram relacionar as duas coisas e a gente nota muita facilidade neles na Matemática em função disso [sic]”.

Com relação ao código ACETA, que representou a aceitação dos alunos percebida pelos professores, foi possível observar que houve um aumento significativo de ocorrências, já que os professores experienciaram, nesta etapa, a aplicação de conceitos em sala de aula. E tiveram oportunidade de interagir com os alunos observando como estes responderam aos estímulos gerados pela utilização do *Scratch* e dos conceitos do Pensamento Computacional.

Corroborando o fato de que a aceitação, por parte dos alunos, foi significativa, em valores absolutos, houve somente uma única ocorrência do código REJEA, detectada na fala da professora P23: “... a maior dificuldade é a questão de tu tirar eles da zona de conforto porque a gente já percebe há bastante tempo na escola que essa geração de alunos que tem hoje na escola estão muito mais voltados para a questão do entretenimento na computação na informática do que para desenvolver pensamento lógico, raciocínio lógico e pensamento computacional [sic]”.

Referindo-se ao fato de que, em um primeiro momento, os alunos pensam na utilização do computador apenas como uma mídia de entretenimento, não conhecendo ou percebendo, muitas vezes, toda a capacidade do computador em favor do ensino.

Foi perceptível, conforme conclusão da professora P23, que a atividade com os alunos gerou bons resultados: “mas no fim a gente conseguiu perceber que a maioria teve entendimento do que a gente estava fazendo e os resultados foram bem legais, bem positivos”.

Destaca-se também o aumento percentual de 150% dos códigos CNTDS e DIFIC, demonstrando, em relação ao código CNTDS, que os professores tiveram problemas para encaixar as atividades do Pensamento Computacional no que concerne aos conteúdos formais que precisam ser desenvolvido nas escolas.

Assim como demonstrado no trecho da fala das professoras P15/P46: “não estava de encontro [sic] com planos de trabalho de ambas disciplinas ...” e P05/P09: “... porém não estamos colocando em prática pois temos muitos conteúdos e trabalhos paralelos para desenvolver ...”.

Ou seja, mesmo com a intenção dos professores em utilizar o que aprenderam du-

⁹² Transcrição de áudio gravado durante entrevista com os professores, da mesma forma que foi realizado e apresentado na seção 4.1.2.2.

rante a formação em Pensamento Computacional, colocando os conhecimentos adquiridos em prática na sala de aula, foi complexo conseguir encaixar atividades adicionais devido à quantidade de conteúdos formais que precisavam ser desenvolvidos com os alunos.

O que justifica, por exemplo, o fato de que na escola DL, conforme seção 4.2.3, as professoras tiveram que realizar as atividades de Pensamento Computacional dentro da carga horária relativa à disciplina de Matemática. O que, pela previsão da BNCC, deveria acontecer naturalmente, já que os conteúdos do Pensamento Computacional estão na área da Matemática.

Porém, da forma como foi exposto pelas professoras, parece que ter que utilizar as aulas de Matemática para realizar a oficina, junto com o pesquisador, foi uma forma de resolver o problema de não ter um horário na grade escolar específico para este fim e não uma tentativa de integrar as atividades do Pensamento Computacional nas aulas de Matemática.

Com relação ao código DIFIC, novamente se configurou o fato de que os professores de Matemática demonstraram estar menos confortáveis com a utilização dos conceitos discutidos, durante a formação, em comparação aos professores de Informática, uma vez que todas as ocorrências do código DIFIC, nesta segunda etapa, foram geradas analisando as falas dos professores de Matemática (P00, P05, P09, P21 e P39).

Com destaque para a fala da professora P21: “...os alunos já estão muito mais avançados que eu, não sei se eu estou no nível deles”. Em relação a esta fala em específico, o pesquisador já havia comentado com o grupo de professoras que este tipo de situação é perfeitamente normal, e devemos estar preparados para isto.

Durante a ocorrência do curso de formação, o pesquisador havia comentado sobre uma ocasião em que ministrava uma aula sobre programação com crianças, na faixa etária dos 12 anos, e havia preparado um exercício que deveria consumir determinado tempo de atenção da turma.

Contudo, o pesquisador foi surpreendido quando um dos alunos resolveu o problema proposto em um tempo muito inferior ao planejado, com uma solução que se julgou muito mais “inteligente” do que aquela proposta como solução inicial.

No contexto tecnológico em que vivemos, situações como essa são comumente vivenciadas pelos professores. Neste sentido, é importante que os educadores estejam preparados para integrar as contribuições dos estudantes em suas práticas em sala de aula.

Como destacado por [Azarfam, Asghar e Jabbari \(2012, p. 454\)](#): “Isso não leva os alunos a pensar que seus professores não estão no controle da sala de aula. Os alunos adoram quando podem compartilhar seus conhecimentos com outras pessoas, especialmente com os professores”.

Ainda com relação ao código DIFIC, destacou-se o relato das professoras de Matemática P05/P09: “Insegurança, por não dominar boa parte do Pensamento Computacional”. O relato das professoras P05/P09 foi alinhado com a realidade encontrada durante o curso de formação, no qual, mesmo com acompanhamento do pesquisador (que deu explicações, solucionou dúvidas e forneceu estruturas de apoio que permitiam aos professores se apropriar dos conceitos do Pensamento Computacional e utilização do *Scratch*) ocorreu ao final a conclusão de alguns professores de Matemática de que “faltava domínio...”, “não conheciam...”, “não eram capazes...”.

E estas falas dos professores sobre dificuldades, assim como foi discutido na seção 4.1.4.2, foram preocupantes, pois representaram um fator limitante já que se esperava dos professores o papel de disseminadores do Pensamento Computacional junto aos alunos.

Esta situação é muito similar ao que descreveu⁹³ Bingimlas (2009, p. 238): “Portanto, a falta de competência dos professores pode ser uma das fortes barreiras à integração de tecnologias na educação. Também pode ser um dos fatores envolvidos na resistência à mudança”.

Referindo-se ao fato de que, ao admitir a carência em sua competência técnica, o professor, ao invés de buscar uma evolução deste quesito, acaba por criar uma resistência à formação tecnológica.

Realizando ainda uma comparação com o trabalho⁹⁴ citado, o mesmo descreveu a falta de tempo como uma das barreiras ao desenvolvimento dos professores quanto à tentativa de utilizar os computadores em sala de aula: “[...] muitos professores têm competência e confiança no uso de computadores [...] mas ainda fazem pouco uso das tecnologias porque não têm tempo suficiente”.

Observando que uma situação cuja falta de tempo foi citada, na escola BJ, onde a fala da professora P21 estava totalmente alinhada com esta realidade: “Depende também da escola continuar liberando este espaço e estes horários pra gente [sic]”, com a fala da professora⁹⁵ referindo-se ao laboratório onde foi realizada a oficina de Pensamento Computacional e aos horários que foram disponibilizados pela direção da escola. Horários que eram inicialmente utilizados para o reforço de Matemática.

De forma complementar ao que foi apresentado na figura 30, a tabela 8 mostra a quantidade de ocorrência dos códigos, destacando a resposta dos professores para cada um dos dois momentos observados.

As células com conteúdo em negrito, na tabela 8, representam os códigos⁹⁶ que

⁹³ Minha tradução para o texto original: “Hence, lack of teacher competence may be one of the strong barriers to the integration of technologies into education. It may also be one of the factors involved in resistance to change”.

⁹⁴ Ibidem. p. 239.

⁹⁵ Que foi pontuada no código HORAP.

⁹⁶ O código HABIP não foi listado pelo fato de que não teve nenhuma ocorrência para os dois momentos

Tabela 8 – Comparação de ocorrências dos códigos pré/pós visitação nas escolas.

Cód.	ETAPA DE FORMAÇÃO										T 1	ETAPA DE VISITAÇÃO NAS ESCOLAS										T 2	T1 + T2
	P 0 0	P 0 5	P 0 9	P 1 5	P 2 1	P 2 2	P 2 3	P 3 7	P 3 9	P 4 6		P 0 0	P 0 5	P 0 9	P 1 5	P 2 1	P 2 2	P 2 3	P 3 7	P 3 9	P 4 6		
	ACETP			X	X		X	X	X				5	X	X	X		X	X	X	X		
ACETA	X		X		X	X					4		X	X	X	X		X	X	X	X	8	12
INTER	X		X	X	X		X	X			6					X		X	X	X		4	10
IMPLE	X	X		X	X			X			5	X				X	X		X			4	9
CNTDS		X	X								2		X	X	X	X		X			X	5	7
DIFIC			X					X			2	X	X	X		X				X		5	7
GRADE	X	X		X			X				4				X					X		2	6
HORAP					X	X		X			3					X	X			X		3	6
ADEQO	X			X				X			3				X					X		2	5
DEFIC		X			X						2					X			X	X		3	5
NUMAL				X	X	X		X			4					X						1	5
PRFAX	X	X	X								3		X	X								2	5
HABAL					X			X			2				X						X	2	4
HORAL					X						1					X	X			X		3	4
REJEP								X			1		X	X						X		3	4
RESPP							X				1		X	X								2	3
LEGIS		X	X								2											0	2
REJEA											0							X				1	1

Fonte: elaborada pelo autor.

coincidiram nas falas de cada professor na etapa de formação e na etapa de visitação às escolas. Sobrepondo as ocorrências de códigos nos dois momentos, observou-se que, para o código ACETP, não houve ocorrência deste código nas falas da professora P46, ou seja, 90% dos professores citaram a aceitação da formação em suas falas.

E para o código ACETA, após sobreposição das ocorrências deste código, nos dois momentos, verificou-se que todos os professores citaram a aceitação dos alunos. Estes resultados trouxeram uma boa perspectiva com relação à aceitação da formação em um contexto geral.

No que diz respeito ao código DIFIC, foi possível observar que, na etapa de formação, havia duas ocorrências, sendo uma da Matemática e outra da Informática. Já na etapa de visitação nas escolas, foram cinco ocorrências deste código - todas da Matemática.

Ou seja, mesmo com os indícios de aceitação geral nas falas dos professores, permaneceram fragmentos da resistência dos professores da Matemática.

O código HORAP foi particularmente importante nas falas das professoras P21 e P22 (escola BJ), nos dois momentos de análise dos códigos, refletindo, perfeitamente, a situação presenciada pelo pesquisador durante a visita na escola.

Com o código HORAP relacionado com a carga horária disponibilizada para as avaliados.

professoras, trabalharem com atividades relacionadas à informática (neste caso, em específico, com relação à oficina de Pensamento Computacional).

Ainda, o código HORAL, que, na primeira etapa da análise havia sido detectado somente na fala de P21, na segunda etapa, não só pôde ser detectada a menção da mesma professora como também para P22.

Esta situação pode ser explicada pelo fato de que a professora P22, quando foi entrevistada na segunda etapa, havia se envolvido diretamente com a organização da oficina de Pensamento Computacional e acabou por vivenciar a dificuldade de disponibilizar horários do laboratório da escola.

O código PRFAX apresentou na primeira etapa três ocorrências, sendo as mesmas geradas pelas falas das professoras de Matemática P00, P05 e P09, porém, na segunda etapa, estas ocorrências foram somente de P05 e P09.

Demonstra-se que, para a professora P00, que foi atendida pelo colega da Informática P23 na escola ST, a necessidade do professor auxiliar foi suprida. Já para as professoras da escola DL, que não tiveram participação da colega da Informática, suas falas geraram novamente a ocorrência do código PRFAX.

Realmente, conforme observado pelo pesquisador na escola DL, não houve um trabalho interdisciplinar da Matemática com a Informática. E esta situação atestou claramente a forma como os professores de Matemática encararam a formação, assumindo que a responsabilidade por disseminar o Pensamento Computacional seria dos professores de Informática, tanto que, na escola DL, onde a professora de Informática não interveio junto com as professoras de Matemática, nenhuma atividade foi realizada antes da visita do pesquisador na escola, ao contrário dos demais estabelecimentos de ensino visitados.

A dificuldade das professoras P05 e P09 também pôde ser percebida observando-se que, na segunda etapa de geração de códigos, as mesmas pontuaram o código REJEP, que se referiu a qualquer situação associada com rejeição dos professores em relação à formação.

Este código, na primeira etapa havia, sido detectado somente na fala de P37, com referência à professora P39, que não havia demonstrado interesse em participar da formação de maneira interdisciplinar. Este fato realmente se confirmou na segunda etapa, quando, inclusive, a fala de P39 pontuou o código REJEP.

A comparação das ocorrências de códigos, referentes à etapa de formação dos professores, com a etapa de visitação nas escolas trouxe mais indícios sobre a resistência dos professores de Matemática com relação à formação de uma forma geral.

Foi possível observar que as escolas que tiveram propostas de atividades, previamente à visita do pesquisador, contavam com um professor de Informática que as condu-

ziu, no entanto, na escola onde a professora de Informática não atuou, as professoras de Matemática não conseguiram propor ações neste sentido.

4.4 Análise dos Programas

A etapa de formação dos professores permitiu que estes tivessem contato com os principais conceitos do Pensamento Computacional e com a utilização do *Scratch*. Durante esta etapa, três programas foram desenvolvidos⁹⁷ para permitir que os docentes exercitassem os conceitos que foram discutidos e que pudessem ser avaliados.

Os programas desenvolvidos pelos professores foram analisados com a ferramenta *Dr. Scratch*, que forneceu uma pontuação para os programas analisados em relação aos conceitos do Pensamento Computacional.

Neste seção da tese, foi realizada uma avaliação dos programas que os professores produziram durante a etapa de visitaç o nas escolas para utilizar com os seus alunos, em sala de aula, e tamb m dos programas produzidos por aqueles estudantes.

O prop sito desta avalia o foi verificar se existiram e quais foram as rela es entre o desempenho dos professores e o correspondente desempenho dos alunos.

A tabela 9 apresenta informa es referentes   pontua o dos programas criados pelos professores e pelos alunos, onde, resumidamente, podem ser observadas as pontua es m dias de cada ator, em cada escola, a pontua o total relativa a cada conceito do Pensamento Computacional e o valor de correla o referente a esta pontua o.

Estas informa es foram compiladas dos dados apresentados no ap ndice F, onde constam todos os programas produzidos e pontua es correspondentes.

Tabela 9 – Pontua o do *Dr. Scratch* para programas criados pelos professores e alunos.

Escola		Conceito do PC							M�dias									
BJ	Professores	L P n e o n i b r o s c t r r a � v. s o z. o	�	a	r	e	p.	t.	c	t	16							
	Alunos										14.3							
DL	Professores										9							
	Alunos										8.8							
LS	Professores										5							
	Alunos										7.5							
ST	Professores										5.5							
	Alunos										5.7							
JR	Professores										13							
	Alunos										10.3							
Pontua�o total por conceito	Professores										8	7	14	13	14	8	6	Correla�o = 0.94
	Alunos										36	40	95	76	95	25	40	

Fonte: elaborada pelo autor.

⁹⁷ Cf. se o 4.1.7.

Com a análise dos dados apresentados na tabela 9, foi possível observar que, para as maiores médias apresentadas para a pontuação de programas dos professores, houve maior média na pontuação apresentada pelos alunos.

Esta análise das pontuações comprovou um comportamento observado pelo pesquisador durante as atividades realizadas nas escolas: nos casos em que os professores desenvolveram suas atividades, previamente à intervenção do pesquisador, os alunos conseguiram construir conceitos de maneira mais consistente.

E, desta forma, estes alunos desenvolveram programas mais bem elaborados, os quais obtiveram uma maior pontuação quando avaliados na ferramenta *Dr. Scratch*.

Observou-se também que os dois conceitos do Pensamento Computacional, com maior pontuação, foram “Interatividade com o Usuário” e “Controle de fluxo”, cujo resultado, referente ao conceito de interatividade com o usuário, apresentado nesta tese, sendo similar ao citado por⁹⁸ Kalogiannakis e Papadakis (2017) sobre os programas criados nos *workshops* propostos, também apresentando altos níveis de componentes de interação com o usuário.

Uma justificativa para este resultado foi o fato de haver sido empregado o formato de jogos na maioria dos programas criados junto com os alunos e professores nas escolas visitadas.

E os jogos criados no *Scratch* utilizam comandos do *mouse* e do teclado que pontuam o conceito de Interatividade com o Usuário no *Dr. Scratch*.

Em relação aos pontos específicos relacionados a cada conceito do Pensamento Computacional, foi realizado o cálculo da correlação entre a pontuação dos professores e a pontuação dos alunos, cujo valor calculado de 0,94 representa uma correlação positiva forte. A interpretação matemática se deu pelo fato de que as maiores pontuações dos conceitos dos alunos foram muito similares, assim como as maiores pontuações dos conceitos dos professores.

Foi possível, também, inferir que os conceitos mais dominados pelos professores foram aqueles mais utilizados durante as suas atividades com os alunos.

E, conseqüentemente, os alunos acabaram por utilizar os conceitos influenciados diretamente pela forma como os professores mediarão as atividades.

Com relação às médias apresentadas para as pontuações dos programas, observou-se que a escola com o maior valor foi a BJ, seguida pela escola JR. Nestas escolas, os professores trabalharam de forma independente e autônoma em relação à intervenção do pesquisador⁹⁹, onde as professoras de Matemática e de Informática demonstraram que trabalharam de forma interdisciplinar e integrada.

⁹⁸ Cf. seção 3.1.1.

⁹⁹ Cf. seções 4.2.2 e 4.2.6.

Na escola DL, o trabalho desenvolvido foi criado durante a intervenção do pesquisador junto às duas professoras de Matemática, obtendo uma pontuação superior às escolas ST e LS, sendo este um dado interessante que trouxe também um indício de que os melhores resultados apresentados ocorreram quando as atividades foram conduzidas em conjunto pela Matemática e Informática.

Com relação à escola LS, o trabalho desenvolvido foi efetuado durante a intervenção do pesquisador junto à professora de Informática, porém, a mesma não teve um suporte interdisciplinar da professora de Matemática.

E na escola ST, o professor de Informática já havia utilizado o *Scratch* junto aos alunos, porém, a ação das professoras de Matemática só ocorreu junto à intervenção do pesquisador. Contudo, as docentes não foram além de simples espectadoras, diferentemente do que foi observado nas escolas BJ e JR.

Além disso, as pontuações médias apresentadas na tabela 9 reforçaram que os melhores resultados ocorreram quando houve ação conjunta das professoras de Matemática e Informática, de forma independente à intervenção do pesquisador com as duas maiores médias sendo obtidas nas escolas BJ e JR.

Mereceu destaque a pontuação obtida por um aluno da escola JR, que pontuou 19 pontos, de 21 possíveis, sendo esta a pontuação maior entre todas as pontuações de alunos e professores em todas as escolas. Com esta pontuação do aluno, tornou-se possível compreender que o papel mais importante do professor foi o de introduzir os conceitos e deixar o aluno “voar” sozinho depois.

Esta situação permitiu uma comparação com o que trouxe [Yadav et al. \(2014\)](#)¹⁰⁰ sobre a importância de o aluno ser um indivíduo ativo na produção de tecnologia.

Ainda, analisando as pontuações dos programas, na escola LS, a média de pontuação da professora foi de 5 pontos, porém, ocorreu uma pontuação de um aluno obtendo 11 pontos, de 21 possíveis, e com uma média de pontuação dos alunos de 5,7.

Ou seja, mesmo com uma pontuação mais baixa do professor, a turma conseguiu uma pontuação média mais alta, inclusive com um aluno chegando a 11 pontos.

Comprovando que o papel do professor foi o de introduzir o assunto, não sendo necessário, muitas vezes, o domínio total do mesmo, tornando mais importante o papel de mediador, similarmente ao que traz [Prado \(2005, p. 15\)](#):

A mediação do professor é fundamental, pois, ao mesmo tempo que o aluno precisa reconhecer sua própria autoria no projeto, ele também precisa sentir a presença do professor, que ouve, questiona e orienta, visando propiciar a construção de conhecimento do aluno.

¹⁰⁰ Cf. seção 3.1.

Após analisar o desempenho de professores trabalhando com seus alunos, foi possível tomar como apropriado o papel do professor descrito¹⁰¹ por Eloy et al. (2017, p. 443):

Os professores participantes do projeto piloto não eram especialistas na área, mas sim pessoas interessadas em aprender algo novo. Em todo o processo, o principal aprendizado descrito por eles foi o novo papel que tiveram na sala de aula: mentores e facilitadores que não têm a resposta para todas as perguntas e quem poderia aprender com e com seus alunos.

Outra análise importante, trazida por¹⁰² Prado (2005), indicou que o professor se depara (no atual contexto tecnológico) com uma situação que implica em novas aprendizagens.

Em consequência disso, requiere-se que haja uma mudança em sua prática pedagógica. Com os exemplos que foram citados acima, podendo-se considerar evidência de situações nas quais os professores demonstraram a chamada “mudança” em suas práticas. Frente aos resultados apresentados pelos alunos, poder-se-ia ousar a conclusão de que tais resultados foram positivos.

Também, pode-se questionar a ideia¹⁰² de Curzon et al. (2014) sobre a necessidade de um conhecimento “profundo” do professor que conduz uma iniciativa sobre o Pensamento Computacional. Não que um bom conhecimento da área seja dispensável, mas em um contexto ainda incipiente, como foi o da introdução ao Pensamento Computacional, mesmo em casos onde os professores não possuíam um grande domínio do Pensamento Computacional e do *Scratch*, os alunos demonstraram um bom aproveitamento do que lhes foi oferecido.

Realizando uma comparação¹⁰³ com o trabalho de Romero, Lepage e Lille (2017) em relação às pontuações apresentadas pelos participantes, 91 de um total de 107 obtiveram 6 pontos na análise do *Dr. Scratch* após desenvolverem programas utilizando conceitos oferecidos durante as atividades realizadas na pesquisa.

Já a pontuação mínima obtida para os programas entregues pelos alunos, relatados nesta tese, foi de 5 pontos, pontuação máxima de 19 pontos e pontuação média de 9,5 pontos.

Assim, obtivemos um resultado que poderia ser associado com mais conceitos do Pensamento Computacional sendo compreendidos pelos alunos e, conseqüentemente, permitindo-lhes criar programas que obtiveram maior pontuação.

¹⁰¹ Minha tradução para o texto original: *The teachers participating in the pilot were not experts in the field, but rather people interested in learning something new. Throughout the process, the main learning described by them was the new role they had in the classroom: mentors and facilitators who do not have the answer to all questions and who could learn from and with their students.*

¹⁰² Cf. seção 3.1.

¹⁰³ Cf. seção 4.1.7.

E uma vez que os conceitos que foram assimilados pelos alunos sobre o Pensamento Computacional, utilizados nos programas criados no *Scratch*, e obtidos durante a atividade com os professores, conclui-se que a formação atingiu resultados positivos.

Considerações sobre o Capítulo

Neste capítulo, foi apresentada a metodologia da tese na qual foram descritas as etapas relativas ao curso de formação dos professores e às atividades que foram realizadas na etapa de visitaç o nas escolas.

Foram realizadas duas an lises de conte do, sendo a primeira relativa aos dados gerados pelas falas dos professores, durante o curso de forma o, e a segunda relativa aos dados gerados pelas falas dos professores durante a visita o nas escolas.

Ainda, foram analisados os dados gerados referentes aos programas criados pelos professores (nas etapas de forma o e visita o nas escolas) e pelos alunos utilizando o *Scratch*.

As an lises de conte do demonstraram que, em um contexto geral, o curso de forma o permitiu aos professores se apropriarem de um conhecimento b sico dos conceitos do Pensamento Computacional, da mesma forma que os instruiu na utiliza o do *Scratch* e seus blocos b sicos de programa o, sendo totalmente percept vel que os professores da Matem tica foram os que se sentiram menos confort veis durante a forma o.

Mas, mesmo apresentando uma resist ncia aos conceitos apresentados, conseguiram desenvolver as tarefas propostas com poucas exce es. Quando foram acompanhados pelo pesquisador em sala de aula, mesmo que n o tenham atuado como protagonistas das atividades desenvolvidas, participaram das mesmas e compreenderam sua utilidade.

Pode-se afirmar, desta forma, que os docentes obtiveram  xito em compartilhar, pelo menos, as ideias b sicas discutidas na forma o.

Os professores de Inform tica, assim como esperado, agiram de forma mais ativa, e este fato foi comprovado pela descri o da atua o dos mesmos pelos seus colegas da Matem tica.

Nas escolas onde os professores de Inform tica “conduziram” seus colegas da Matem tica, comprovou-se, quantitativamente, pela an lise da pontua o dos programas, que os resultados apresentados foram melhores. Tanto os resultados dos professores, quanto os resultados dos alunos.

Alguns n meros referentes   etapa metodol gica podem ser citados:

- Foram 49 professores que participaram da etapa de forma o e que receberam certifica o;

- Um total de 50 horas de atividades foram contabilizadas na etapa de formação;
- O pesquisador visitou 7 escolas, participando de atividades com os professores e alunos totalizando uma carga horária¹⁰⁴ de 25 horas de atividades;
- Na etapa de formação, foram criados 115 programas¹⁰⁵ pelos professores;
- Na etapa de visitação nas escolas, foram criados 56 programas¹⁰⁵ sendo 7 programas desenvolvidos pelos professores/pesquisador e 49 pelos alunos.

Em comparação com alguns dos trabalhos apresentados nos capítulos 2 e 3, foi possível observar que os professores - que foram sujeitos da pesquisa desta tese - apresentaram características particulares, já que foram selecionados professores de Matemática e Informática.

Pode-se apontar como exemplo [Silva, Silva e França \(2017\)](#), cujos autores pesquisaram professores com formação pedagógica e tecnológica sem especificar suas áreas do conhecimento.

Em [Ramos e Espadeiro \(2014\)](#) foram pesquisados professores dos cursos de formação inicial de Educação Básica e de mestrado para os Ensinos Básico e Secundário de Artes Visuais, Biologia, Geologia, Educação Física, Matemática, Português, Espanhol e Francês.

No trabalho de [Eloy et al. \(2017\)](#), os professores sujeitos da pesquisa eram de diversas áreas, mas com uma representatividade de 41% de professores da área da Matemática.

Nesta tese, participaram da formação 49 professores, sendo 37 professores de Matemática (75,5%) e 12 professores de Informática (25,5%). Avaliar os professores de Matemática foi alinhado ao fato de que os conhecimentos relacionados ao Pensamento Computacional foram associados com a área da Matemática na [BNCC \(2019\)](#), e, conseqüentemente, os professores desta área deveriam ser os responsáveis por difundir os conhecimentos do Pensamento Computacional.

Ainda, mesclar os professores da Matemática com os da Informática permitiu avaliar a real situação da estrutura das escolas do município onde foi conduzida a pesquisa desta tese. Pois, devido a esta constatação, a maioria das escolas conta com um professor de Informática que trabalha interdisciplinarmente com os demais professores.

Também foi importante realizar uma comparação desta tese com relação às etapas metodológicas realizadas nos trabalhos apresentados, observando que em [Ramos e Espadeiro \(2014\)](#), [Silva, Silva e França \(2017\)](#), [Bower et al. \(2017\)](#), [Romero, Lepage e Lille](#)

¹⁰⁴ As reuniões com professores e direção foram dimensionadas para 50 minutos de duração e as atividades em sala de aula foram dimensionadas para duas horas de duração.

¹⁰⁵ Todos os programas criados foram compartilhados com a comunidade de *Scratchers*.

(2017), Kalogiannakis e Papadakis (2017) e Yadav, Gretter e McLean (2017) as pesquisas foram conduzidas com professores em formação e fora do ambiente escolar.

O fato de avaliar os professores fora de um ambiente escolar, durante atuação com alunos, parece sempre trazer resultados positivos nas conclusões dos autores, pois permitem afirmações generalizadas como a de que os professores tiveram um aumento em determinada capacidade relacionada ao Pensamento Computacional, ou conseguiram definir o termo Pensamento Computacional após a formação.

Mas a ausência de uma etapa onde os professores realmente tenham aplicado em sala de aula os conhecimentos junto a uma turma de alunos parece não expressar a realidade que se costuma encontrar.

A etapa metodológica, desta tese, por avaliar os participantes durante uma etapa de formação, e em uma etapa posterior, trouxe resultados com forte relação à atuação dos professores em sala de aula. E as análises realizadas foram mais alinhadas com a etapa metodológica de Eloy et al. (2017).

Porém, mesmo com etapas metodológicas similares, há uma visão mais crítica conforme relatos sobre o desempenho dos professores em sala de aula.

A comparação desta tese com os trabalhos apresentados também mostrou que a escolha do *Scratch* como ferramenta para realizar atividades plugadas do Pensamento Computacional foi adotada por vários autores. Assim como foi relatado nos trabalhos de Silva, Silva e França (2017), Eloy et al. (2017), Bower et al. (2017) e Romero, Lepage e Lille (2017).

Da mesma forma que a utilização do *Dr. Scratch*, como ferramenta de avaliação, foi citada por Romero, Lepage e Lille (2017).

5 Conclusões e Trabalhos Futuros

O Objetivo Geral definido nesta tese foi: “Compreender como os professores de Matemática e Informática dos Anos Finais do ensino básico se apropriam dos conhecimentos de um curso de formação em Pensamento Computacional, aplicando estes conhecimentos em atividades de sala de aula”.

A revisão da literatura realizada enfatiza o quão fundamental é o docente estar continuamente atualizado e procurando reinventar-se, permitindo-lhe estabelecer a importância do papel do professor para, assim, poder exercê-lo como um agente de transformação. Porém, há uma grande lacuna entre a teoria e a prática - conforme discutido em vários aspectos sobre a carência da formação dos professores -, principalmente no contexto tecnológico alinhado à pesquisa realizada nesta tese.

Nunca a necessidade de atualização do professor ficou tão evidente como em nosso cenário atual, e, ao propor o contato dos professores de Informática e Matemática com os conceitos do Pensamento Computacional e com o *Scratch*, esta tese permitiu a coleta de dados em uma situação em que os docentes precisaram trabalhar diretamente com a tecnologia, agregando um novo conhecimento (Pensamento Computacional) àqueles já utilizados em suas disciplinas. Mas, ao analisarmos os resultados apresentados, especialmente das análises de conteúdo, verificou-se o quanto foi difícil para os docentes se apropriarem dos novos conhecimentos.

Assim como a constatação de que os professores tiveram dificuldades em se apropriar dos conhecimentos relativos ao Pensamento Computacional, constatou-se que o sistema educacional como um todo parece ainda não estar preparado para adaptar-se às necessidades do novo milênio.

Tópicos como inovação, empreendedorismo e tecnologia são amplamente discutidos nas mídias, nos círculos educacionais e nas empresas, as quais precisam que os alunos de hoje, futuros profissionais do amanhã, sejam formados para a nova realidade. No entanto, a adaptação do sistema educacional para atender estas demandas caminha em um ritmo descompassado àquele que seria necessário.

Ao interagir com os professores que foram sujeitos de pesquisa desta tese, foi possível perceber que os problemas que enfrentam no seu dia a dia são os mais variados. Por exemplo: a infraestrutura da totalidade das escolas que foram visitadas, por mais que a grande maioria daquelas instituições de ensino possuíssem laboratórios funcionais, apresentavam equipamentos tecnologicamente ultrapassados.

As velocidades das conexões com a *Internet* não atendiam à demanda de ambientes

online, como o *Scratch*, e não existia um setor de TI (Tecnologia da Informação) ou similar para fornecer um suporte ativo aos laboratórios. Estas observações sobre infraestruturas das escolas corroboram Eloy et al. (2017, p. 443):

Infraestrutura: esse é um grande desafio nas escolas públicas Brasileiras. Embora as escolas participantes tenham laboratórios de informática, os professores ainda relatam problemas relacionados à qualidade dos computadores e conexão à *Internet*.

Alguns professores de Matemática apresentaram dificuldades em relação aos itens básicos de Informática como criar usuário e senha para utilizar o *Scratch*, no modo *online*, demonstrando, assim, que não seria tão simples que conseguissem assimilar os conceitos do Pensamento Computacional e como utilizar mesmo que de forma básica o *Scratch*. Mas, de qualquer forma, os mesmos, em sua grande maioria, tentaram realizar as tarefas propostas e conseguiram apresentar resultados satisfatórios.

Retomando os objetivos específicos propostos nesta tese, podemos verificar como cada um deles foi atendido. Iniciando com o item de avaliar o desempenho dos professores em relação à utilização dos conceitos do Pensamento Computacional/*Scratch*, durante a etapa de formação, verificou-se que os 49 professores que foram certificados conseguiram produzir pelo menos um dos programas propostos.

Ainda que os programas que foram produzidos pelos professores não fossem complexos, os mesmos apresentaram ao menos uma dimensão básica do Pensamento Computacional como a lógica, concluindo, deste modo, que os professores conseguiram pelo menos um desempenho “básico¹” ao utilizar o *Scratch* durante a etapa de formação.

Com relação ao objetivo de avaliar como os professores utilizaram os conceitos do Pensamento Computacional/*Scratch in loco* nas escolas após a etapa de formação, os resultados apresentados demonstraram diferentes reações e empenho por parte dos professores para aplicar os conhecimentos adquiridos. E os melhores resultados apresentados foram aqueles que demonstraram iniciativa por parte dos professores previamente à atuação do pesquisador e onde os professores de Informática atuaram junto com os de Matemática.

Para o objetivo de identificar as diferenças na percepção sobre a formação e nas atividades realizadas em sala de aula, de acordo com a formação do professor (Informática/Matemática), desde a etapa de formação, os resultados já demonstravam que os professores de Matemática apresentaram maior resistência em assimilar os conhecimentos do Pensamento Computacional e do *Scratch*.

Apesar de os professores de Matemática terem apresentado desempenhos satisfatórios na criação de propostas com o *Scratch*, em nenhum momento tiveram iniciativa

¹ Utilizando aqui o termo básico assim como proposto na classificação do *Dr. Scratch*.

de produzir/propor sozinho atividades com os alunos. Já os professores da Informática protagonizaram a condução das atividades de um modo geral, guiando os colegas da Matemática e criando propostas funcionais que foram utilizadas com os alunos.

Com relação ao Objetivo Geral de compreender como os professores de Matemática e Informática dos anos finais do ensino básico se apropriaram dos conhecimentos do curso de formação em Pensamento Computacional, aplicando-os em atividades de sala de aula, verificou-se que, pelo menos nas escolas BJ e JR, os resultados apresentados puderam ser considerados positivos.

Os professores das escolas BJ e JR se apropriaram dos conhecimentos do Pensamento Computacional, pelo menos os básicos, como a lógica, e conseguiram propor atividades que foram utilizadas pelos alunos. Estes estudantes conseguiram utilizar os conteúdos transmitidos por seus professores, comprovadamente, adotando como métrica de avaliação as pontuações geradas na análise do *Dr. Scratch*. Vale observar que, nas escolas destacadas, houve uma atuação interdisciplinar das professoras de Matemática com as professoras de Informática. Com a interdisciplinaridade e o perfil das professoras sendo um fator importante para os resultados obtidos.

Foi possível verificar que mesmo com as professoras de Informática conduzindo na maior parte do tempo as atividades do Pensamento Computacional/*Scratch*, as professoras de Matemática possuíam uma boa compreensão do que havia sido criado e conseguiam esclarecer dúvidas básicas dos alunos, como, por exemplo, explicar o funcionamento de um bloco de programação no *Scratch*.

Mas, em um contexto geral, a forma como os professores se apropriaram dos conhecimentos do Pensamento Computacional/*Scratch* pareceu superficial, pois, após o período de formação, poucas ações foram efetivamente implementadas nas escolas visitadas.

Com exceção da escola BJ (onde a proposta da oficina de Pensamento Computacional ocorreu desde o período de formação, até o momento da visita do pesquisador) e da escola JR (onde as atividades propostas se desenvolveram anteriormente à visita do pesquisador e ocorreu quantitativamente o segundo melhor resultado com os alunos), foi constatada situação diferente nas demais escolas.

Na escola DL, não houve ação por parte das professoras de Matemática, com o agravante de que as mesmas não conseguiram estabelecer uma relação interdisciplinar com a professora de Informática daquele estabelecimento educacional.

Na escola LS, a professora de Informática conduziu as ações sozinha, sem ter a participação ativa da professora de Matemática, enquanto na escola ST, o professor de Informática conduziu ações com os alunos, porém, as professoras de Matemática somente participaram quando houve visita do pesquisador. Além disso, é fundamental citar que, nestas escolas, nenhuma ação foi realizada pelas professoras de Matemática, individual-

mente, ou seja, somente foram criadas propostas ou em conjunto com os professores de Informática, ou que tenham aguardado pela atuação do pesquisador durante a realização das atividades.

Desta forma, considerando esta amostra de professores que foram sujeitos de pesquisa desta tese, parece que a proposta da BNCC (2019) de associar o Pensamento Computacional à área da Matemática seja de difícil implementação. Ainda, o material, desenvolvido e apresentado nesta tese sobre conceitos do Pensamento Computacional e atividades plugadas com o *Scratch*, pode ser reproduzido com professores de Matemática, Informática e das outras áreas de conhecimento, contribuindo para ampliar a literatura como material disponível para formação dos professores em Pensamento Computacional, alinhado à necessidade apresentada na BNCC.

Uma vez que os melhores resultados apresentados nas escolas visitadas foram originados por grupos onde professores das duas formações trabalharam de forma interdisciplinar (como nas escolas BJ e JR), estes dados trouxeram uma contribuição significativa para a tese. Diante disso, considerar um time interdisciplinar de professores (como, por exemplo, da Matemática + Informática) poderia ser uma alternativa viável para implementar ações do Pensamento Computacional em sala de aula.

Apesar dos resultados significativos apresentados pelos professores nas escolas visitadas, foi perceptível que as ações tomadas pareceram como simples atendimento à visita do pesquisador e não como ações permanentes inseridas no currículo das disciplinas. Este tipo de situação poderá ser um entrave ao estabelecimento de ações permanentes com o intuito de implantar o Pensamento Computacional no contexto escolar, com vistas aos preceitos preconizados pela BNCC.

Ainda, a falta de uma carga horária apropriada para realizar as atividades e a própria relação que existiu entre a Matemática e a Informática, nas escolas do município, pareceu não ter contribuído para resultados satisfatórios em todas as escolas visitadas.

Como limitação desta tese, cita-se que o grupo de professores que foi pesquisado não foi formado com base em outros critérios de seleção além de possuir a formação em Matemática ou Informática. Desta forma, trabalhos futuros poderiam formar um grupo utilizando critérios, tal como selecionar professores de outras matérias (Ciências, História e Português), verificando através de uma pré-triagem aqueles que tivessem uma afinidade com tecnologia e teriam, assim, maior tendência a assimilar os conhecimentos do Pensamento Computacional.

Além do mais, destaca-se que aumentar o número de escolas visitadas e, consequentemente, de professores avaliados em sala de aula, traria uma visão mais abrangente da etapa de pós- formação. Porém, limitação de recursos e de tempo hábil do pesquisador não permitiram alongar esta etapa da pesquisa. Logo, trabalhos futuros poderiam padro-

nizar as duas etapas apresentadas na metodologia desta tese e aplicar pesquisa similar na rede estadual de educação.

Destaca-se que esta tese, frente aos trabalhos similares com os quais realizaram-se comparações, assim como Curzon et al. (2014) e Bower et al. (2017), trouxe uma contribuição adicional. Além da etapa de formação dos professores, ocorreu uma etapa de visitação e de atividades realizadas nas escolas, permitindo avaliar como os professores trabalharam os conhecimentos, dos quais se apropriaram na formação, com seus alunos em sala de aula.

Sugere-se para trabalhos futuros que ambientes similares ao *Scratch* sejam testados quando for adotada uma abordagem plugada. Por exemplo: o *Scratchx* (<http://scratchx.org/>) traz uma série de “extensões” do *Scratch* que permitem fazer conexões com *Hardware*s como o *Arduino* e criar programas que convertem texto para voz.

Outro ambiente que segue a mesma lógica de programação por blocos que o *Scratch* é o *kitten*², que oferece funções avançadas, inclusive com blocos de inteligência artificial.

Contextualizar conceitos do Pensamento Computacional, como a abstração, poderia trazer resultados diferenciados, utilizando elementos como botões e sensores, ou elementos de robótica associados com o *Arduino*, ampliando os horizontes da pesquisa.

² <https://kitten.code.game/>

Referências

- ABRUCIO, F. L. *Formação de professores no Brasil: diagnóstico, agenda de políticas e estratégias para a mudança*. São Paulo: Editora Moderna, 2016. Citado 2 vezes nas páginas 55 e 61.
- ALENCAR, A. F. e Wilkerson Andrade e R. Pensamento computacional em sala de aula: Desafios, possibilidades e a formação docente. *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, v. 4, n. 1, p. 1226 – 1235, 2015. ISSN 2316-8889. Disponível em: <<https://www.br-ie.org/pub/index.php/wcbie/article/view/6262>>. Citado na página 66.
- ALMEIDA, M. E. B. de; VALENTE, J. A. Pensamento computacional nas políticas e nas práticas em alguns países. *Revista Observatório*, v. 5, n. 1, p. 202–242, 2019. ISSN 2447-4266. Disponível em: <<https://sistemas.uft.edu.br/periodicos/index.php/observatorio/article/download/4742/14697/>>. Acesso em: 20/03/2019. Citado 2 vezes nas páginas 98 e 105.
- AMARAL, H. J. C. do et al. A disseminação do pensamento computacional na educação básica: lições aprendidas com experiências de licenciandos em computação. In: *XXXIV Congresso da Sociedade Brasileira de Computação CSBC 2014*. SBC - Sociedade Brasileira de Computação, 2014. p. 1473–1482. Disponível em: <<https://s3.amazonaws.com/academia.edu.documents/34359195/A-disseminacao-do-pensamento-computacional-na-educacao-basica-licoes-aprendidas.pdf>>. Acesso em: 9.9.2018. Citado 2 vezes nas páginas 35 e 37.
- ARAÚJO, A. L.; ANDRADE, W.; SEREY, D. Pensamento computacional sob a visão dos profissionais da computação: uma discussão sobre conceitos e habilidades. In: *Anais dos Workshops do IV Congresso Brasileiro de Informática na Educação (CBIE 2015)*. SBC - Sociedade Brasileira de Computação, 2015. p. 1454–1463. ISSN 2316-8889. Disponível em: <<http://www.br-ie.org/pub/index.php/wcbie/article/view/6329>>. Acesso em: 9.9.2017. Citado na página 34.
- ARMONI, M.; BEN-ARI, M. *Computer Science Concepts in Scratch (Supplement for Scratch 2.0) Version 1.0*. online, 2013. Disponível em: <<https://stwww1.weizmann.ac.il/scratch/wp-content/uploads/sites/7/2016/04/scratch-20-supplement-1-0-two-side.pdf>>. Citado na página 49.
- ARMONI, M.; MEERBAUM-SALANT, O.; BEN-ARI, M. From scratch to “real” programming. *Trans. Comput. Educ.*, ACM, New York, NY, USA, v. 14, n. 4, p. 25:1–25:15, fev. 2015. ISSN 1946-6226. Disponível em: <<http://doi.acm.org/10.1145/2677087>>. Citado na página 47.
- AZARFAM, Y.; ASGHAR, A.; JABBARI, Y. Dealing with teachers’ technophobia in classroom. v. 2, p. 452–455, 07 2012. Disponível em: <https://www.researchgate.net/publication/277830848_Dealing_with_Teachers'_Technophobia_in_Classroom>. Acesso em: 10.02.2020. Citado na página 131.

BARCELOS, T. S.; SILVEIRA, I. F. Teaching computational thinking in initial series an analysis of the confluence among mathematics and computer sciences in elementary education and its implications for higher education. In: *2012 XXXVIII Conferencia Latinoamericana En Informatica (CLEI)*. Medellín, Colombia: Piscataway: IEEE, 2012. p. 1–8. Citado na página 71.

BARCELOS, T. S.; SILVEIRA, I. F. Relações entre o pensamento computacional e a matemática através da construção de jogos digitais. In: SBC (Ed.). *Proceedings of SBGames 2013*. SBC - Sociedade Brasileira de Computação, 2013. p. 52–55. Disponível em: <<http://www.sbgames.org/sbgames2013/proceedings>>. Acesso em: 17.4.2018. Citado 2 vezes nas páginas 31 e 71.

Barefoot. *Abstraction*. 2014. Disponível em: <<https://barefootcas.org.uk/wp-content/uploads/2014/10/Abstraction-Concept-Barefoot-Computing.pdf>>. Acesso em: 08.04.2018. Citado na página 39.

BASOGAIN, X. et al. Computational thinking in pre-university blended learning classrooms. *Computers in Human Behavior*, p. 1–8, 05 2017. Disponível em: <https://www.researchgate.net/publication/316627086_Computational_Thinking_in_pre-university_Blended_Learning_classrooms>. Acesso em: 08.12.2018. Citado na página 115.

BECKER, F. Modelos pedagógicos e modelos epistemológicos. *Educação e Realidade*, v. 19, p. 89–96, 01 1999. Citado na página 77.

BELL, T.; WITTEN, I. H.; FELLOWS, M. *Ensinando Ciência da Computação sem o uso do computador - Tradução coordenada por Luciano Porto Barreto*. Computer Science Unplugged ORG, 2011. Disponível em: <<https://classic.csunplugged.org/wp-content/uploads/2014/12/CSUnpluggedTeachers-portuguese-brazil-feb-2011.pdf>>. Citado 2 vezes nas páginas 44 e 45.

BERTO, L.; ZAINA, L.; SAKATA, T. Metodologia para ensino do pensamento computacional para crianças baseada na alternância de atividades plugadas e desplugadas. *Revista Brasileira de Informática na Educação*, v. 27, n. 02, p. 01, 2019. ISSN 2317-6121. Citado na página 46.

BINGIMLAS, K. Barriers to the successful integration of ict in teaching and learning environments: A review of the literature. *Eurasia Journal of Mathematics Science Technology Education*, v. 5, p. 235–245, 08 2009. Disponível em: <<https://www.ejmste.com/download/barriers-to-the-successful-integration-of-ict-in-teaching-and-learning-environments-a-review-of-the-4156.pdf>>. Acesso em: 9.02.2020. Citado 2 vezes nas páginas 59 e 132.

BITESIZE. *KS3 - Introduction to computational thinking*. 2015. Publicação online. Disponível em: <<https://www.bbc.com/bitesize/guides/zp92mp3/revision/1>>. Acesso em: 10.04.2018. Citado na página 38.

BLIKSTEIN, P. *O pensamento computacional e a reinvenção do computador na educação*. 2008. Publicação online. Disponível em: <http://www.blikstein.com/paulo/documents/online/ol_pensamento_computacional.html>. Acesso em: 11.09.2017. Citado na página 33.

- BNCC. *Base Nacional Comum Curricular*. 2019. Publicação *online*. Disponível em: <http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf>. Acesso em: 26.12.2019. Citado 6 vezes nas páginas 36, 71, 98, 121, 140 e 145.
- BOE, B. et al. Hairball: Lint-inspired static analysis of scratch projects. In: *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*. New York, NY, USA: ACM, 2013. (SIGCSE '13), p. 215–220. ISBN 978-1-4503-1868-6. Disponível em: <<http://doi.acm.org/10.1145/2445196.2445265>>. Citado na página 46.
- BONDAR, M. Prehistoric innovations: Wheels and wheeled vehicles. *Acta Archaeologica Academiae Scientiarum Hungaricae*, v. 69, p. 271–297, 12 2018. Citado na página 26.
- BOWER, M. et al. Improving the computational thinking pedagogical capabilities of school teachers. *Australian Journal of Teacher Education*, Social Science Press, v. 42, n. 3, p. 53–72, 2017. ISSN 0313-5373. Citado 8 vezes nas páginas 30, 63, 69, 94, 106, 140, 141 e 146.
- BRACKMANN, C. P. *Desenvolvimento do Pensamento Computacional Através de Atividades Desplugadas na Educação Básica*. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre - RS, 2017. Citado 3 vezes nas páginas 30, 45 e 98.
- BRENNAN, K.; RESNICK, M. New frameworks for studying and assessing the development of computational thinking. In: *Proceedings of 2012 Annual Meeting Program*. Washington, DC, USA: AERA - American Educational Research Association, 2012. p. 1–25. ISSN 0163-9676. Citado 3 vezes nas páginas 47, 65 e 117.
- CALAO, L. A. et al. Developing mathematical thinking with scratch. In: _____. *Design for Teaching and Learning in a Networked World: 10th European Conference on Technology Enhanced Learning, EC-TEL 2015, Toledo, Spain, September 15-18, 2015, Proceedings*. Cham: Springer International Publishing, 2015. p. 17–27. ISBN 978-3-319-24258-3. Disponível em: <https://doi.org/10.1007/978-3-319-24258-3_2>. Citado na página 47.
- CLEMENTS, D. H. Computers in early childhood mathematics. *Contemporary Issues in Early Childhood*, v. 3, n. 2, p. 160–181, 2002. Disponível em: <<https://doi.org/10.2304/ciec.2002.3.2.2>>. Citado na página 43.
- CODE. *UNPLUGGED Computational Thinking*. 2013. Publicação *online*. Disponível em: <<https://code.org/curriculum/course3/1/Teacher>>. Acesso em: 12.04.2018. Citado 2 vezes nas páginas 39 e 40.
- COIMBRA, C. L. Os Modelos de Formação de Professores/as da Educação Básica: quem formamos? *Educação Realidade*, scielo, v. 45, 00 2020. ISSN 2175-6236. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S2175-62362020000100604nrm=iso>. Citado na página 54.
- CRONBACH, L. J. Coefficient alpha and the internal structure of tests. *Psychometrika*, v. 16, n. 3, p. 297–334, Sep 1951. ISSN 1860-0980. Disponível em: <<https://doi.org/10.1007/BF02310555>>. Citado na página 84.

- CSIZMADIA, A. et al. *Computational thinking - a guide for teachers*. London, 2015. Disponível em: <<https://eprints.soton.ac.uk/424545/>>. Citado 3 vezes nas páginas 35, 43 e 61.
- CURZON, P. et al. Enthusiasing & inspiring with reusable kinaesthetic activities. *SIGCSE Bull.*, ACM, New York, NY, USA, v. 41, n. 3, p. 94–98, jul. 2009. ISSN 0097-8418. Disponível em: <<http://doi.acm.org/10.1145/1595496.1562911>>. Citado na página 97.
- CURZON, P. et al. Introducing teachers to computational thinking using unplugged storytelling. In: *Proceedings of the 9th Workshop in Primary and Secondary Computing Education*. New York, NY, USA: Association for Computing Machinery, 2014. (WiPSCE '14), p. 89–92. ISBN 9781450332507. Disponível em: <<https://doi.org/10.1145/2670757-2670767>>. Citado 5 vezes nas páginas 60, 61, 113, 138 e 146.
- CYSNEIROS, P. G. *Professores e máquinas: uma concepção de informática na educação*. 1996. Publicação online. Disponível em: <<https://www.academia.edu/3015011>>. Acesso em: 19.10.2019. Citado na página 57.
- Digital Promise. *Computational Thinking for a Computational World*. 2017. Publicação online. Disponível em: <<https://digitalpromise.org/wp-content/uploads/2017/12/dp-comp-thinking-v1r5.pdf>>. Acesso em: 01.02.2018. Citado 3 vezes nas páginas 62, 72 e 108.
- DODERO, J. M.; MOTA, J. M.; RUIZ-RUBE, I. Bringing computational thinking to teachers' training: A workshop review. In: *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality*. New York, NY, USA: Association for Computing Machinery, 2017. (TEEM 2017). ISBN 9781450353861. Disponível em: <<https://doi.org/10.1145/3144826.3145352>>. Citado na página 44.
- ECHALAR, A. D. L. F.; PEIXOTO, J. Programa Um Computador por Aluno: o acesso às tecnologias digitais como estratégia para a redução das desigualdades sociais. *Ensaio: Avaliação e Políticas Públicas em Educação*, scielo, v. 25, p. 393 – 413, 04 2017. ISSN 0104-4036. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0104-40362017000200393nrm=iso>. Citado na página 59.
- ELOY, A. A. d. S. et al. Programming literacy: Computational thinking in brazilian public schools. In: *Proceedings of the 2017 Conference on Interaction Design and Children*. New York, NY, USA: ACM, 2017. (IDC '17), p. 439–444. ISBN 978-1-4503-4921-5. Disponível em: <<http://doi.acm.org/10.1145/3078072.3084306>>. Citado 7 vezes nas páginas 65, 70, 113, 138, 140, 141 e 143.
- ERNI, M. et al. Computational thinking - a tool to motivate understanding in elementary school teachers. In: . Lisboa, Portugal: Springer International, 2016. v. 583, p. 348–364. ISBN 978-3-319-29584-8. Citado na página 44.
- FAGUNDES, T. B. Os conceitos de professor pesquisador e professor reflexivo: perspectivas do trabalho docente. *Revista Brasileira de Educação*, scielo, v. 21, p. 281 – 298, 06 2016. ISSN 1413-2478. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S1413-24782016000200281nrm=iso>. Citado na página 54.

- FAZENDA, I. C. A. *Integração e interdisciplinaridade no ensino brasileiro - Efetividade ou ideologia*. São Paulo: Edições Loyola, 1979. (Coleção "Realidade educacional"). ISBN 9788515005062. Citado na página 99.
- FERREIRA, A. C. C. et al. Experiência prática interdisciplinar do raciocínio computacional em atividades de computação desplugada na educação básica. In: *Anais do XXI Workshop de Informática na Escola (WIE 2015)*. Maceió, AL, Brasil: SBC, 2015. p. 256–265. Disponível em: <<http://br-ie.org/pub/index.php/wie/article/view/5032>>. Citado na página 28.
- FERREIRA, A. de A.; SILVA, B. D. da. Comunidade de prática *on-line*: uma estratégia para o desenvolvimento profissional dos professores de História, *journal = Educação em Revista*. scielo, v. 30, p. 37 – 64, 03 2014. ISSN 0102-4698. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0102-46982014000100003nrm=iso>. Citado na página 83.
- FILHO, D. B. F.; JÚNIOR, J. A. da S. Desvendando os mistérios do coeficiente de correlação de pearson (r). *Revista Política Hoje*, Universidade Federal de Pernambuco (UFPE), v. 18, p. 115 – 146, 1 2009. Disponível em: <<https://periodicos.ufpe.br/revistas/politicohoje/article/viewFile/3852/3156>>. Acesso em: 23/01/2019. Citado na página 110.
- FRANÇA, R.; TEDESCO, P. Desafios e oportunidades ao ensino do pensamento computacional na educação básica no brasil. In: *Anais dos Workshops do IV Congresso Brasileiro de Informática na Educação (CBIE 2015)*. Sociedade Brasileira de Computação, 2015. p. 1464 – 1473. Disponível em: <<https://www.br-ie.org/pub/index.php/wcbie/article/view/6331>>. Acesso em: 5.5.2019. Citado na página 37.
- FREIRE, P. Carta de Paulo Freire aos professores. *Estudos Avançados*, scielo, v. 15, p. 259–268, 08 2001. ISSN 0103-4014. Disponível em: <<http://www.scielo.br/pdf/ea/v15n42/v15n42a13.pdf>>. Acesso em: 02.10.2018. Citado 2 vezes nas páginas 53 e 54.
- FUSTE, A.; SCHMANDT, C. Hypercubes: A playful introduction to computational thinking in augmented reality. In: *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*. New York, NY, USA: Association for Computing Machinery, 2019. (CHI PLAY '19 Extended Abstracts), p. 379–387. ISBN 9781450368711. Disponível em: <<https://doi.org/10.1145/3341215-3356264>>. Citado 2 vezes nas páginas 46 e 116.
- GABINI, W. S. A.; DINIZ, R. E. A. d. S. A FORMAÇÃO CONTINUADA, O USO DO COMPUTADOR E AS AULAS DE CIÊNCIAS NOS ANOS INICIAIS DO ENSINO FUNDAMENTAL. *Ensaio Pesquisa em Educação em Ciências (Belo Horizonte)*, scielo, v. 14, p. 333 – 348, 12 2012. ISSN 1983-2117. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S1983-21172012000300333nrm=iso>. Citado na página 113.
- GARRISON, D. et al. Revisiting methodological issues in transcript analysis: Negotiated coding and reliability. *The Internet and Higher Education*, v. 9, n. 1, p. 1–8, 2006. ISSN 1096-7516. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1096751605000771>>. Citado na página 94.

GATTI, B. A. Formação de professores no Brasil: características e problemas. *Educação Sociedade*, scielo, v. 31, p. 1355 – 1379, 12 2010. ISSN 0101-7330. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0101-73302010000400016nrm=iso>. Citado 2 vezes nas páginas 55 e 61.

GEORGE, D.; MALLERY, P. *SPSS for Windows Step-by-Step: A Simple Guide and Reference, 15.0 Update*. 8th. ed. Needham Heights, MA, USA: Allyn & Bacon, Inc., 2007. ISBN 0205569072, 9780205569076. Citado na página 85.

GERHARDT, T. E.; SILVEIRA, D. T. *Métodos de Pesquisa*. 1^a. ed. Plageder, 2009. (Série Educação a Distância - UFRGS). ISBN 9788538600718. Disponível em: <<https://books.google.com.br/books?id=dRuzRyEIzmkC>>. Citado na página 85.

GOULART, C. M. A. O conceito de letramento em questão: por uma perspectiva discursiva da alfabetização. *Bakhtiniana, Rev. Estud. Discurso*, São Paulo, v. 9, n. 2, p. 35 – 51, Dec. 2014. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S2176-45732014000200004lng=enrm=iso>. Citado na página 35.

GROVER, S.; PEA, R. Computational thinking in k-12: A review of the state of the field. *Educational Researcher*, v. 42, n. 1, p. 38–43, 2013. Disponível em: <<https://doi.org/10.3102/0013189X12463051>>. Citado 3 vezes nas páginas 41, 42 e 44.

HU, C. Computational thinking: What it might mean and what we might do about it. In: *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*. New York, NY, USA: ACM, 2011. (ITiCSE '11), p. 223–227. ISBN 978-1-4503-0697-3. Disponível em: <<http://doi.acm.org/10.1145/1999747.1999811>>. Citado na página 60.

IMBERMAN, S.; STURM, D.; AZHAR, M. Q. Computational thinking: expanding the toolkit. v. 29, p. 39–46, 06 2014. Citado na página 37.

Instituto Ayrton Senna. *Formação continuada de professores no Brasil: Acelerando o desenvolvimento dos nossos educadores*. 2014. Publicação online. Disponível em: <https://www.observatoriodopne.org.br/_uploads/_posts/46.pdf?1140547125>. Acesso em: 11.09.2018. Citado na página 54.

ISTE. *How to develop computational thinkers*. 2018. Publicação online. Disponível em: <<https://www.iste.org/explore/articleDetail?articleid=2137category=Computational-Thinkingarticle=>>>. Acesso em: 20.05.2018. Citado 2 vezes nas páginas 39 e 40.

JACOMINI, M. A.; PENNA, M. G. d. O. Carreira docente e valorização do magistério: condições de trabalho e desenvolvimento profissional. *Pro-Posições*, scielo, v. 27, p. 177 – 202, 08 2016. ISSN 0103-7307. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0103-73072016000200177nrm=iso>. Citado 2 vezes nas páginas 58 e 59.

JÚNIOR, S. D. da S.; COSTA, F. J. Mensuração e escalas de verificação: uma análise comparativa das escalas de likert e *Phrase Completion*. *PMKT - Revista Brasileira de Pesquisas de Marketing, opinião e Mídia*, v. 15, p. 1–16, Junho 2014. ISSN 23170-0123. Disponível em: <<http://http://sistema.semead.com.br/17semead/resultado/trabalhosPDF/1012.pdf>>. Citado na página 84.

KALOGIANNAKIS, M.; PAPADAKIS, S. A proposal for teaching scratchjr programming environment in preservice kindergarten teachers. In: *Proceedings of Research, Practice and Collaboration in Science Education - ESERA 2017*. Dublin City University, 2017. p. 2095–2105. Disponível em: <<https://www.academia.edu/34914458>>. Citado 3 vezes nas páginas 63, 136 e 141.

KAZIMOGLU, C. et al. Learning programming at the computational thinking level via digital game-play. *Procedia Computer Science*, v. 9, p. 522 – 531, 2012. ISSN 1877-0509. Proceedings of the International Conference on Computational Science, ICCS 2012. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1877050912001779>>. Citado na página 43.

KINDERGARTEN, G. L. *SCRATCH*. 2007. Página da WEB. Disponível em: <<https://scratch.mit.edu/about>>. Acesso em: 12.10.2017. Citado na página 46.

KOHTALA, C. *Making Sustainability: How Fab Labs Address Environmental Sustainability*. 185 p. Tese (Doutorado), 03 2016. Citado na página 29.

KONG, S. C.; ABELSON, H. *Computational Thinking Education*. Springer Nature, 2019. ISBN 9789811365287. Disponível em: <<https://library.oapen.org/handle/20.500-12657/23182>>. Citado na página 37.

LEITE, M. et al. Pensamento computacional nas escolas: Limitado pela tecnologia, infraestrutura ou prática docente? In: *Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação (CBIE 2017)*. Recife, PE: SBC - Sociedade Brasileira de Computação, 2017. p. 1002–1010. Citado 4 vezes nas páginas 30, 95, 96 e 98.

LOCKWOOD, J.; MOONEY, A. Computational thinking in education: Where does it fit? a systematic literary review. *CoRR*, abs/1703.07659, Mar. 2017. Disponível em: <<https://arxiv.org/abs/1703.07659>>. Acesso em: 06.11.2018. Citado na página 36.

MA, W. W.-k.; ANDERSSON, R.; STREITH, K.-O. Examining user acceptance of computer technology: an empirical study of student teachers. *Journal of Computer Assisted Learning*, Blackwell Publishing Ltd, v. 21, n. 6, p. 387–395, 2005. ISSN 1365-2729. Disponível em: <<http://dx.doi.org/10.1111/j.1365-2729.2005.00145.x>>. Citado 3 vezes nas páginas 84, 96 e 99.

Magna Publications. *Using Badges in the Classroom to Motivate Learning*. 2014. Publicação online. Disponível em: <<http://www.facultyfocus.com/articles/teaching-with-technology-articles/using-badges-classroom-motivate-learning/>>. Acesso em: 21.06.2019. Citado na página 115.

MANDAJI, M. et al. O programaê! e a formação de professores para a integração do pensamento computacional ao currículo. In: *Anais CIET:EnPED:2018 – Educação e Tecnologias: Docência e mediação pedagógica*. São Carlos - SP: UFSCar, 2018. Citado na página 61.

MEERBAUM-SALANT, O.; ARMONI, M.; BEN-ARI, M. M. Learning computer science concepts with scratch. In: *Proceedings of the Sixth International Workshop on Computing Education Research*. New York, NY, USA: ACM, 2010. (ICER '10), p. 69–76. ISBN 978-1-4503-0257-9. Disponível em: <<http://doi.acm.org/10.1145/1839594.1839607>>. Citado na página 105.

MELLO, G. N. d. Formação inicial de professores para a educação básica: uma (re)visão radical. *São Paulo em Perspectiva*, scielo, v. 14, p. 98 – 110, 03 2000. ISSN 0102-8839. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0102-88392000000100012nrm=iso>. Citado 2 vezes nas páginas 54 e 127.

MIOTTO, P.; CARDOSO, V. C. A utilização do software scratch para o ensino e a aprendizagem do conceito de função. *Os Desafios da Escola Pública Paranaense na Perspectiva do Professor PDE*, I, 2014. Disponível em: <http://www.diaadiaeducacao.pr.gov.br/portals/cadernospde/pdebusca/producoes_pde/2014/2014_uem_mat_artigo_polyana_miotto.pdf>. Citado 2 vezes nas páginas 30 e 100.

MIRANDA, M. do R. A. C. *O IMPACTO DA DESMOTIVAÇÃO NO DESEMPENHO DOS PROFESSORES*. 168 p. Dissertação (Mestrado) — Universidade Católica Portuguesa, Porto, 2012. Citado na página 106.

MOHAGHEGH, M.; MCCAULEY, M. Computational thinking: The skill set of the 21st century. *International Journal of Computer Science and Information Technology*, v. 7, p. 1524–1530, 06 2016. Disponível em: <https://www.researchgate.net/profile/Mahsa_Mohaghegh/publication/303792583_Computational_Thinking_The_Skill_Set_of_the_21st_Century/links/5752960f08ae17e65ec37669/Computational-Thinking-The-Skill-Set-of-the-21st-Century.pdf?origin=publication_detail>. Citado 2 vezes nas páginas 29 e 30.

MONKS, J.; SCHMIDT, R. M. The Impact of Class Size on Outcomes in Higher Education. *The B.E. Journal of Economic Analysis & Policy*, v. 11, n. 1, p. 1–19, September 2011. Disponível em: <<https://ideas.repec.org/a/bpj/bejeap/v11y2011i1n62.html>>. Citado na página 72.

MORAIS, A. D. d.; BASSO, M. V. d. A.; FAGUNDES, L. A. d. C. Educação Matemática Ciência da Computação na escola: aprender a programar fomenta a aprendizagem de matemática? *Ciência Educação (Bauru)*, scielo, v. 23, p. 455 – 473, 06 2017. ISSN 1516-7313. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S1516-73132017000200455nrm=iso>. Citado 3 vezes nas páginas 30, 47 e 98.

MORAN, J. *Novas Tecnologias E Mediação Pedagógica*. Papirus, 2000. (Coleção Papirus Educação). ISBN 9788530805944. Disponível em: <https://books.google.com.br/books?id=i7uhwQM_PyEC>. Citado na página 56.

MORENO-LEON, J.; ROBLES, G.; ROMAN-GONZALEZ, M. Dr. scratch: Automatic analysis of scratch projects to assess and foster computational thinking. *RED. Revista de Educación a Distancia*, Universidad de Murcia, n. 46, p. 1–23, 2015. Citado 3 vezes nas páginas 42, 50 e 102.

MORENO-LEON, J.; ROBLES, G.; ROMAN-GONZALEZ, M. Towards data-driven learning paths to develop computational thinking with scratch. *IEEE Transactions on Emerging Topics in Computing*, v. 16, Aug 2017. ISSN 2168-6750. Disponível em: <<http://ieeexplore.ieee.org/abstract/document/7999170/>>. Citado 2 vezes nas páginas 48 e 50.

- M.Z, N. et al. Defining the concept of innovation and firm innovativeness: A critical analysis from resource-based view perspective. *International Journal of Business and Management*, v. 11, p. 87–94, 05 2016. Citado na página 29.
- NETO, J. J. S. et al. Uma escala para medir a infraestrutura escolar. *Estudos em Avaliação Educacional*, v. 24, n. 54, p. 78 – 99, jan 2013. Citado na página 96.
- Nova Escola. *Os caminhos para a formação de professores*. 2009. Publicação online. Disponível em: <<https://gestaoescolar.org.br/conteudo/772/os-caminhos-para-a-formacao-de-professores>>. Acesso em: 29.02.2018. Citado na página 89.
- OLIVEIRA, E. C. de L. *O uso do Scratch no ensino fundamental: possibilidades de incorporação curricular segundo professoras dos anos iniciais*. Dissertação (Mestrado) — Pontifícia Universidade Católica de Minas Gerais, Belo Horizonte, 2009. Citado 3 vezes nas páginas 30, 47 e 95.
- PAPERT, S. *Mindstorms: Children, Computers, and Powerful Ideas*. USA: Basic Books, Inc., 1980. ISBN 0465046274. Citado na página 33.
- PAPERT, S. *A máquina das crianças: repensando a escola na era da informática*. Artmed, 2008. ISBN 9788536310589. Disponível em: <<https://books.google.com.br/books?id=jGHFMwAACAAJ>>. Citado na página 89.
- PAZ, L. A. S. do C. O pensamento computacional e a formação continuada de professores: uma experiência com as tics. *Revista on line de Política e Gestão Educacional*, v. 21, n. 1, p. 1655–1677, 2017. ISSN 1519-9029. Disponível em: <<https://periodicos.fclar.unesp.br/rpge/article/view/10095>>. Citado 2 vezes nas páginas 65 e 72.
- PENNSYLVANIA, W. U. of. *What if Computers Become Smarter Than Humans?* 2016. Publicação online. Disponível em: <<http://knowledge.wharton.upenn.edu/article/will-superhuman-artificial-intelligence-turn-us-paper-clips/>>. Acesso em: 20.11.2018. Citado na página 34.
- PENTEADO, M. et al. *A informática em ação: formação de professores, pesquisa e extensão*. 1ª. ed. São Paulo, SP: Olho d'Água, 2000. Citado 3 vezes nas páginas 97, 101 e 112.
- PERRENOUD, P. Formar professores em contextos sociais em mudança: prática reflexiva e participação crítica. *Revista Brasileira de Educação*, Campinas, v. 12, p. 5 – 21, 9 – 12 1999. Disponível em: <https://www.unige.ch/fapse/SSE/teachers/perrenoud/php_main/php_1999/1999_34.html>. Citado na página 58.
- PINTO, J. A. M. d. R. Uma análise do financiamento da educação no estado da Califórnia, EUA. *Cadernos de Pesquisa*, scielo, v. 35, p. 699 – 722, 12 2005. ISSN 0100-1574. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0100-15742005000300009nrm=iso>. Citado na página 46.
- PRADO, M. E. B. B. *Pedagogia de projetos: fundamentos e implicações*. 2005. Disponível em: <<http://portal.mec.gov.br/seed/arquivos/pdf/1sf.pdf>>. Acesso em: 08.09.2018. Citado 3 vezes nas páginas 56, 137 e 138.

PROTTSMAN, K. *Computational Thinking Meets Student Learning: Extending the ISTE Standards*. International Society for Technology in Education, 2019. ISBN 9781564847614. Disponível em: <<https://books.google.com.br/books?id=RhbRvwEACAAJ>>. Citado na página 38.

RAMOS, J. L.; ESPADEIRO, R. Os futuros professores e os professores do futuro. os desafios da introdução ao pensamento computacional na escola, no currículo e na aprendizagem. *Educação, Formação Tecnologias*, v. 7, n. 2, 2014. ISSN 1646-933X. Disponível em: <<https://eft.educom.pt/index.php/eft/article/view/462>>. Citado 3 vezes nas páginas 65, 72 e 140.

REINALDO, F. et al. Uso de smartphones na educação: Avaliação por grupos focais. Universidade Lusófona do Porto, Porto, Portugal, v. 1 (2016): Atas - Investigação Qualitativa em Educação, p. 769–778, 2016. Disponível em: <<https://proceedings.ciaiq.org/index.php/ciaiq2016/article/view/668/657>>. Citado na página 112.

REPENNING, A. *Computational Thinking \neq Programming*. 2016. Publicação online. Disponível em: <<https://magazine.swissinformatics.org/en/computational-thinking-%E2%89%A0-programming/>>. Acesso em: 21.11.2017. Citado 2 vezes nas páginas 41 e 43.

RESNICK, M. et al. Scratch: Programming for all. *Commun. ACM*, ACM, New York, NY, USA, v. 52, n. 11, p. 60–67, nov. 2009. ISSN 0001-0782. Disponível em: <<http://doi.acm.org/10.1145/1592761.1592779>>. Citado na página 46.

Revista Educação. *Quais os desafios dos professores para incorporar as novas tecnologias no ensino*. 2018. Disponível em: <<http://www.revistaeducacao.com.br/quais-os-desafios-dos-professores-para-incorporar-as-novas-tecnologias-no-ensino/>>. Acesso em: 08.09.2018. Citado 2 vezes nas páginas 56 e 101.

RIBEIRO, A. E. Letramento digital: um tema em gêneros efêmeros. *Revista da ABRALIN*, v. 8, n. 1, 2017. ISSN 0102-7158. Disponível em: <<https://revistas.ufpr.br/abralin/article/view/52433>>. Citado na página 35.

RIBEIRO, L.; FOSS, L.; CAVALHEIRO, S. A. da C. *Entendendo o Pensamento Computacional*. 2017. Disponível em: <<https://arxiv.org/pdf/1707.00338.pdf>>. Acesso em: 8.08.2018. Citado 3 vezes nas páginas 38, 39 e 41.

ROCHA, A. K. d. O.; PRADO, M. E. B. B. Uma abordagem tecnológica na formação do professor de matemática. *Revista Tecnologias na Educação*, Ano 6, n. 11, 2014. Citado na página 37.

ROCHA, K. C. da. Programando com o scratch na aula de matemática. *Revista RENOTE - Novas Tecnologias na Educação*, Cinted UFRGS, Porto Alegre, RS, Brasil, v. 13, n. 2, p. 1–10, mar. 2015. ISSN 1679-1916. Disponível em: <<http://seer.ufrgs.br/index.php/renote/article/view/61429/36321>>. Citado na página 47.

RODRIGUES, C. A. F. d. S.; LIMA, F. J. A. C. d.; BARBOSA, F. T. A. Importance of using basic statistics adequately in clinical research. *Revista Brasileira de Anestesiologia*, scielo, Campinas, São Paulo, v. 67, p. 619–625, 12 2017. ISSN 0034-7094. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0034-70942017000600619nrm=iso>. Citado na página 106.

ROMAN, A. F. e Carina Tonieto e M. A formação de professores reflexivos: a docência como objeto de investigação. *Educação (UFSM)*, v. 38, n. 2, p. 277–287, 2013. ISSN 1984-6444. Disponível em: <<https://periodicos.ufsm.br/reeducacao/article/view/5483>>. Citado na página 54.

ROMERO, M.; LEPAGE, A.; LILLE, B. Computational thinking development through creative programming in higher education. *International Journal of Educational Technology in Higher Education*, v. 14, n. 1, p. 42, Dec 2017. ISSN 2365-9440. Disponível em: <<https://doi.org/10.1186/s41239-017-0080-z>>. Citado 5 vezes nas páginas 64, 109, 112, 138 e 141.

SÁ, J. d. S.; WERLE, F. O. C. Infraestrutura escolar e espaço físico em educação: o estado da arte. *Cadernos de Pesquisa*, scielo, v. 47, p. 386 – 413, 06 2017. ISSN 0100-1574. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S0100-15742017000200001nrm=iso>. Citado na página 59.

SALDANA, J. *The Coding Manual for Qualitative Researchers*. SAGE Publications, 2009. ISBN 9781446200124. Disponível em: <<https://books.google.com.br/books?id=msuOE0UfXpUC>>. Citado na página 92.

SCHUHMACHER, V. R. N.; FILHO, J. d. P. A.; SCHUHMACHER, E. As barreiras da prática docente no uso das tecnologias de informação e comunicação. *Ciência e Educação (Bauru)*, scielo, v. 23, p. 563 – 576, 07 2017. ISSN 1516-7313. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S1516-73132017000300563nrm=iso>. Citado 3 vezes nas páginas 31, 59 e 72.

SELBY, C. C. Relationships: Computational thinking, pedagogy of programming, and bloom's taxonomy. In: *Proceedings of the Workshop in Primary and Secondary Computing Education*. New York, NY, USA: ACM, 2015. (WiPSCE '15), p. 80–87. ISBN 978-1-4503-3753-3. Disponível em: <<http://doi.acm.org/10.1145/2818314.2818315>>. Citado na página 74.

SILVA, J. M. e D. A influência da estrutura escolar no processo de ensino-aprendizagem: uma análise baseada nas experiências do estágio supervisionado em geografia. *Geografia Ensino Pesquisa*, v. 19, n. 3, p. 19–28, 2015. ISSN 2236-4994. Disponível em: <<https://periodicos.ufsm.br/geografia/article/view/14315>>. Citado na página 59.

SILVA, V.; SILVA, K.; FRANÇA, R. S. Pensamento computacional na formação de professores: experiências e desafios encontrados no ensino da computação em escolas públicas. In: *Anais do XXIII Workshop de Informática na Escola (WIE 2017) do IV Congresso Brasileiro de Informática na Educação (CBIE 2017)*. SBC - Sociedade Brasileira de Computação, 2017. p. 805–814. Disponível em: <<http://www.br-ie.org/pub/index.php/wie/article/viewFile/7299/5097>>. Acesso em: 9.5.2018. Citado 6 vezes nas páginas 37, 59, 62, 63, 140 e 141.

SOUZA, D. T. R. de. Formação continuada de professores e fracasso escolar: problematizando o argumento da incompetência. *Educação e Pesquisa*, scielo, v. 32, p. 477 – 492, 12 2006. ISSN 1517-9702. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttextpid=S1517-97022006000300004nrm=iso>. Citado na página 101.

- STEMLER, S. An overview of content analysis. *Practical Assessment, Research & Evaluation*, v. 7, n. 17, Jun 2001. Disponível em: <<https://scholarworks.umass.edu/cgi/viewcontent.cgi?article=1100&context=pars>>. Acesso em: 22/03/2018. Citado 2 vezes nas páginas 92 e 94.
- STERLING, L. *Computers may be evolving but are they intelligent?* 2016. Publicação online. Disponível em: <<http://theconversation.com/computers-may-be-evolving-but-are-they-intelligent-58205>>. Acesso em: 20.11.2018. Citado na página 34.
- TALLVID, M. Understanding teachers' reluctance to the pedagogical use of ict in the 1: 1 classroom. *Education and Information Technologies*, Kluwer Academic Publishers, Hingham, MA, USA, v. 21, n. 3, p. 503–519, maio 2016. ISSN 1360-2357. Disponível em: <<http://dx.doi.org/10.1007/s10639-014-9335-7>>. Citado na página 31.
- TELLES, M.; HSIEH, Y. *The Science of Debugging*. USA: Coriolis Group Books, 2001. ISBN 1576109178. Citado na página 43.
- VALENTE, J. Integração do pensamento computacional no currículo da educação básica: Diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. *Revista e-Curriculum*, v. 14, n. 3, p. 864–897, 2016. ISSN 1809-3876. Disponível em: <<https://revistas.pucsp.br/index.php/curriculum/article/view/29051>>. Citado na página 37.
- VALENTE, J. Pensamento computacional, letramento computacional ou competência digital? novos desafios da educação. *Educação e Cultura Contemporânea*, v. 16, p. 147–168, 01 2019. Citado na página 35.
- VIEIRA, A.; PASSOS, O.; BARRETO, R. Um relato de experiência do uso da técnica computação desplugada. In: *XXI WEI*. UFMG, 2013. p. 670–679. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/wei/2013/0031.pdf>>. Citado na página 45.
- VINCENT-LANCRIN, S. et al. *Measuring Innovation in Education 2019*. Paris: OECD Publishing, 2019. 336 p. Disponível em: <<https://www.oecd-ilibrary.org/content/publication/9789264311671-en>>. Citado na página 29.
- WEINTROP, D. et al. Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, Springer Netherlands, v. 25, n. 1, p. 127–147, 2 2016. ISSN 1059-0145. Citado na página 71.
- WENGER, E. *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press, 1999. (Learning in Doing: Social, Cognitive and Computational Perspectives). ISBN 9780521663632. Disponível em: <<https://books.google.com.br/books?id=heBZpgYUKdAC>>. Citado na página 83.
- WING, J. M. Computational thinking. *Commun. ACM*, ACM, New York, NY, USA, v. 49, n. 3, p. 33–35, mar. 2006. ISSN 0001-0782. Disponível em: <<http://doi.acm.org/10.1145/1118178.1118215>>. Citado 2 vezes nas páginas 33 e 34.
- WING, J. M. *Computational Thinking*. 2012. Disponível em: <https://www.microsoft.com/en-us/research/wp-content/uploads/2012/08/Jeanette_Wing.pdf>. Acesso em: 18.03.2018. Citado na página 33.

WOHLIN, C. et al. *Experimentation in Software Engineering*. New York: Springer, 2012. ISBN 978-3-642-29043-5. Citado na página 32.

World Economic Forum. *The Future of Jobs Employment, Skills and Workforce Strategy for the Fourth Industrial Revolution*. 2016. Publicação online. Disponível em: <http://www3.weforum.org/docs/WEF_Future_of_Jobs.pdf>. Acesso em: 10.03.2018. Citado na página 36.

YADAV, A.; GRETTER, S.; MCLEAN, J. G. T. Computational thinking in teacher education. In: _____. *Emerging Research, Practice, and Policy on Computational Thinking, Educational Communications and Technology: Issues and Innovations*. Michigan, USA: Springer International, 2017. p. 205–220. ISBN 978-3-319-52690-4. Citado 3 vezes nas páginas 64, 94 e 141.

YADAV, A. et al. Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education (TOCE)*, v. 14, p. 1–5, 03 2014. Citado 3 vezes nas páginas 37, 60 e 137.

ZANELLA, B. R. D.; LIMA, M. de F. W. P. Refletindo sobre os fatores de resistência no uso das tics nos ambientes escolares. *Scientia cum Industria*, v. 05, p. 78–89, 06 2017. Citado na página 31.

ŠERBEC, I. N.; CERAR, ; ŽEROVNIK, A. Developing computational thinking through games in scratch. In: . University of Ljubljana, 2018. p. 21–30. Disponível em: <<http://pefprints.pef.uni-lj.si/5141/>>. Citado na página 50.

Apêndices

APÊNDICE A – Termo de Consentimento

Termo de Consentimento Livre e Esclarecido

Por meio deste documento, solicitamos sua autorização para divulgar dados de pesquisa relacionados à Comunidade de Prática sobre **“Introdução do Pensamento Computacional em sala de aula utilizando o ambiente Scratch”**, realizada pelo acadêmico Taiser Tadeu Teixeira Barros, como parte de seu curso de Doutorado no Programa de Pós-Graduação em Informática na Educação da Universidade Federal do Rio Grande do Sul, com orientação do professor Eliseo Berni Reategui.

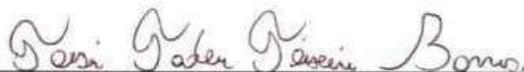
O objetivo da pesquisa é criar um ambiente que propicie aos docentes, compreender conceitos sobre o Pensamento Computacional, capacitando os mesmos a utilizar estes conhecimentos em sala de aula com o intuito de aprimorar sua atividade docente.

A pesquisa será realizada durante encontros presenciais e à distância, no período previsto de Setembro de 2017 à Julho de 2018 (Carga horária estimada de 50 horas/aula). A colaboração dos docentes se dará através de sua participação na Comunidade de Prática, disponível em um ambiente virtual, com fóruns e atividades que permitam a troca de informações entre os participantes e o pesquisador. Qualquer material produzido durante esta interação poderá ser divulgado como resultados da pesquisa, sendo porém garantido o anonimato de cada participante uma vez que a referência (quando necessária) a um ou mais participantes em particular será efetuada por meio de codificação (por exemplo, “...o docente D00 concorda que a aplicação da teoria em sala de aula”).

A participação do docente não é obrigatória e o mesmo tem total liberdade de participar ou não da atividade como um todo ou de atividades em específico às quais o mesmo não considere como relevantes à sua atuação como docente!

Esta pesquisa pretende colaborar com o enriquecimento das estratégias de ensino e de aprendizagem, visando à qualificação da Educação de maneira geral.

Todos os docentes terão um canal de comunicação com o pesquisador para a qualquer momento poder tirar dúvidas, e dar críticas/sugestões que visem contribuir com a atividade!



22 de Setembro de 2017

Taiser Tadeu Teixeira Barros

Assinatura do Pesquisador

Eu,
devidamente esclarecido e concordo em participar da atividade de pesquisa proposta.

..... 22 de Setembro de 2017
Assinatura do Participante

APÊNDICE B – Certificado

**SECRETARIA MUNICIPAL DE
EDUCAÇÃO**

CERTIFICADO

Certificamos que **NOME DA PESSOA** participou da "FORMAÇÃO EM PENSAMENTO COMPUTACIONAL UTILIZANDO SCRATCH", realizada no período de 22 de Setembro de 2017 à 05 de Setembro de 2018.

Carga horária: 50 horas. Aproveitamento: 100%.

..... de de 2018.

 Departamento de Educação
  Secretária Municipal de Educação
 Me. Engº Taiser T. T. Barros
Instrutor

Apoio:






Contato:

CONTEÚDO PROGRAMÁTICO	INSTRUTOR	CARGA HORÁRIA
Apresentação da proposta da formação para os professores e primeiras atividades com o Scratch: criação de usuário e senha, como utilizar os comandos para criar um programa;	Taiser T. T. Barros	4h
Comandos do Scratch e como criar programas para complementar conteúdos desenvolvidos em sala de aula;	Taiser T. T. Barros	4h
Pensamento Lógico: estruturas de decisão SE-ENTÃO e SE-ENTÃO-SENÃO. Interatividade com o usuário: utilizando Bandeira Verde para iniciar o programa, entrada de dados pelo teclado e perguntas ao usuário. Repetir até que, e repetir sempre.	Taiser T. T. Barros	8h
Representação de dados: modificando propriedades de objetos. Utilizando variáveis. Paralelismo: múltiplos atores e múltiplos scripts.	Taiser T. T. Barros	8h
Utilizando os conceitos discutidos para implementar programas que serão utilizados como ferramentas em sala de aula.	Taiser T. T. Barros	8h
Discussão online e desenvolvimento de tarefas.	Taiser T. T. Barros	14h
Encerramento da formação e apresentação de resultados.	Taiser T. T. Barros	4h
Carga Horária Total : 50 horas		






Contato:

APÊNDICE C – Questionário sobre intenção de utilização do computador

Perguntas do questionário.

Item	Descrição
01	Tenho acesso ao computador em sala de aula?
02	Compreendo que meus alunos podem ter conhecimentos de informática iguais ou superiores aos meus e me sinto confortável com esta situação?
03	Usar computadores melhora o desempenho de meu trabalho?
04	Computadores me permitem realizar tarefas mais rapidamente?
05	Usar computadores melhora a minha eficiência no trabalho?
06	Usar computadores melhora a qualidade no trabalho que faço?
07	Usar computadores tornará mais fácil fazer meu trabalho?
08	Raramente me confundo quando uso computadores?
09	Não cometo erros frequentes ao usar computadores?
10	Raramente preciso de ajuda ao usar computadores?
11	Não acho complicado usar computadores?
12	Aprender a operar computadores é fácil para mim?
13	É fácil para mim tornar-me hábil no uso de computadores?
14	Em geral, acho os computadores fáceis de usar?
15	Meus colegas pensam que eu deveria usar computadores?
16	Pessoas que influenciam meu comportamento pensam que eu deveria usar computadores?
17	Meus amigos pensam que eu deveria usar computadores na minha atividade como professor?
18	Pessoas que são importantes para mim pensam que eu deveria usar computadores na minha atividade como professor?
19	Meus superiores (coordenadores/diretores) pensam que eu deveria usar computadores na minha atividade como professor?
20	Sempre que possível, pretendo usar computadores na minha atividade como professor?
21	Na medida do possível, pretendo usar computadores para criar diferentes tarefas de ensino?

APÊNDICE D – Lista dos programas entregues pelos professores durante a formação

Código do Professor	Programa_1	Pontos Dr. Scratch	Programa_2	Pontos Dr. Scratch	Programa_3	Pontos Dr. Scratch
P00	https://scratch.mit.edu/projects/215458066/	6/21	https://scratch.mit.edu/projects/227999062/	10/21	https://scratch.mit.edu/projects/235676509/	5/21
P01	https://scratch.mit.edu/projects/222422878/	5/21	SEM ENTREGA		https://scratch.mit.edu/projects/236311624/	16/21
P02	https://scratch.mit.edu/projects/216873621/	8/21	https://scratch.mit.edu/projects/227180113/	10/21	https://scratch.mit.edu/projects/236320183/	7/21
P03	https://scratch.mit.edu/projects/207293179/	8/21	https://scratch.mit.edu/projects/228956667/	9/21	https://scratch.mit.edu/projects/235687091/	5/21
P04	https://scratch.mit.edu/projects/207291637/	6/21	SEM ENTREGA		SEM ENTREGA	
P05	https://scratch.mit.edu/projects/220119769/	8/21	https://scratch.mit.edu/projects/229293697/	9/21	https://scratch.mit.edu/projects/235687091/	5/21
P06	https://scratch.mit.edu/projects/214017163/	4/21	https://scratch.mit.edu/projects/220680323/	10/21	https://scratch.mit.edu/projects/235676509/	5/21
P07	https://scratch.mit.edu/projects/218189301/	4/21	https://scratch.mit.edu/projects/229203724/	8/21	https://scratch.mit.edu/projects/236320141/	7/21
P08	https://scratch.mit.edu/projects/222343500/	5/21	SEM ENTREGA		SEM ENTREGA	
P09	https://scratch.mit.edu/projects/220415322/	8/21	https://scratch.mit.edu/projects/220681511/	8/21	SEM ENTREGA	
P10	https://scratch.mit.edu/projects/213507236/	5/21	https://scratch.mit.edu/projects/227346334/	10/21	https://scratch.mit.edu/projects/236120027/	5/21
P11	https://scratch.mit.edu/projects/238661024/	6/21	SEM ENTREGA		SEM ENTREGA	
P12	https://scratch.mit.edu/projects/207291637/	6/21	https://scratch.mit.edu/projects/226993930/	13/21	https://scratch.mit.edu/projects/235676623/	9/21
P13	https://scratch.mit.edu/projects/220755998/	6/21	https://scratch.mit.edu/projects/220761710/	9/21	https://scratch.mit.edu/projects/235707846/	9/21
P14	https://scratch.mit.edu/projects/220762862/	3/21	SEM ENTREGA		https://scratch.mit.edu/projects/236337283/	8/21
P15	https://scratch.mit.edu/projects/215439471/	5/21	https://scratch.mit.edu/projects/224872971/	13/21	https://scratch.mit.edu/projects/236337283/	8/21
P16	https://scratch.mit.edu/projects/212188357/	14/21	https://scratch.mit.edu/projects/227436247/	12/21	https://scratch.mit.edu/projects/236337329/	11/21
P17	https://scratch.mit.edu/projects/213733338/	7/21	https://scratch.mit.edu/projects/220779158/	10/21	https://scratch.mit.edu/projects/236360107/	12/21
P18	https://scratch.mit.edu/projects/207360460/	7/21	https://scratch.mit.edu/projects/220761934/	8/21	https://scratch.mit.edu/projects/236343003/	8/21
P19	https://scratch.mit.edu/projects/220765818/	8/21	SEM ENTREGA		https://scratch.mit.edu/projects/236344073/	6/21
P20	https://scratch.mit.edu/projects/217076021/	5/21	https://scratch.mit.edu/projects/227835644/	11/21	https://scratch.mit.edu/projects/235706161/	5/21
P21	https://scratch.mit.edu/projects/214556468/	9/21	https://scratch.mit.edu/projects/225582577/	12/21	https://scratch.mit.edu/projects/236213760/	10/21
P22	https://scratch.mit.edu/projects/214556468/	9/21	https://scratch.mit.edu/projects/220761744/	13/21	https://scratch.mit.edu/projects/236213760/	10/21
P23	https://scratch.mit.edu/projects/214016258/	8/21	https://scratch.mit.edu/projects/220779081/	10/21	https://scratch.mit.edu/projects/236337329/	11/21
P24	https://scratch.mit.edu/projects/217927383/	4/21	https://scratch.mit.edu/projects/229203724/	8/21	https://scratch.mit.edu/projects/235706113/	5/21
P25	https://scratch.mit.edu/projects/206858244/	8/21	https://scratch.mit.edu/projects/226978522/	9/21	https://scratch.mit.edu/projects/236311624/	16/21
P26	https://scratch.mit.edu/projects/217489672/	7/21	https://scratch.mit.edu/projects/226595115/	12/21	https://scratch.mit.edu/projects/236311624/	16/21
P27	https://scratch.mit.edu/projects/216870137/	5/21	SEM ENTREGA		https://scratch.mit.edu/projects/236337443/	9/21
P28	https://scratch.mit.edu/projects/217472899/	9/21	https://scratch.mit.edu/projects/228445561/	10/21	https://scratch.mit.edu/projects/236337489/	10/21
P29	SEM ENTREGA		SEM ENTREGA		SEM ENTREGA	
P30	https://scratch.mit.edu/projects/210461956/	6/21	SEM ENTREGA		https://scratch.mit.edu/projects/236344073/	6/21
P31	https://scratch.mit.edu/projects/217973767/	4/21	https://scratch.mit.edu/projects/222339081/	9/21	https://scratch.mit.edu/projects/236312254/	11/21
P32	https://scratch.mit.edu/projects/239413141/		SEM ENTREGA		SEM ENTREGA	
P33	https://scratch.mit.edu/projects/216351718/	6/21	https://scratch.mit.edu/projects/228450231/	9/21	https://scratch.mit.edu/projects/236312232/	11/21
P34	https://scratch.mit.edu/projects/217157220/	5/21	SEM ENTREGA		SEM ENTREGA	
P36	https://scratch.mit.edu/projects/236320226/	7/21	https://scratch.mit.edu/projects/236653606/	8/21	https://scratch.mit.edu/projects/236312145/	11/21
P37	https://scratch.mit.edu/projects/215995295/	9/21	https://scratch.mit.edu/projects/227508535/	7/21	https://scratch.mit.edu/projects/235679527/	7/21
P38	https://scratch.mit.edu/projects/217319822/	9/21	SEM ENTREGA		https://scratch.mit.edu/projects/235679527/	7/21
P39	https://scratch.mit.edu/projects/216760749/	9/21	SEM ENTREGA		https://scratch.mit.edu/projects/235679527/	7/21
P40	https://scratch.mit.edu/projects/210549479/	6/21	SEM ENTREGA		https://scratch.mit.edu/projects/236337443/	9/21
P41	https://scratch.mit.edu/projects/217310854/	7/21	SEM ENTREGA		SEM ENTREGA	
P42	https://scratch.mit.edu/projects/215581999/	11/21	https://scratch.mit.edu/projects/226290412/	12/21	https://scratch.mit.edu/projects/237569835/	6/21
P43	SEM ENTREGA		https://scratch.mit.edu/projects/222408654/	9/21	SEM ENTREGA	
P44	https://scratch.mit.edu/projects/214158193/	7/21	https://scratch.mit.edu/projects/222408617/	13/21	https://scratch.mit.edu/projects/236337443/	9/21
P45	https://scratch.mit.edu/projects/238246491/	5/21	https://scratch.mit.edu/projects/238246567/	10/21	SEM ENTREGA	
P46	https://scratch.mit.edu/projects/210553516/	6/21	https://scratch.mit.edu/projects/228451942/	10/21	SEM ENTREGA	
P47	https://scratch.mit.edu/projects/218232516/	6/21	SEM ENTREGA		SEM ENTREGA	
P48	SEM ENTREGA		https://scratch.mit.edu/projects/236736278/	9/21	https://scratch.mit.edu/projects/236344073/	6/21
P49	https://scratch.mit.edu/projects/237272847/	11/21	SEM ENTREGA		https://scratch.mit.edu/projects/214994704/	14/21

Observação: As células de código destacadas em cinza com texto em negrito são referentes aos professores de informática e as demais aos professores de matemática.

APÊNDICE E – Pontuação dos programas entregues pelos professores durante a formação

	Conceito do PC														Total
	Lógica		Paralelismo		Interatividade		Representação de dados		Controle de fluxo		Sincronização		Abstração		
	prog1	prog3	prog1	prog3	prog1	prog3	prog1	prog3	prog1	prog3	prog1	prog3	prog1	prog3	
P02	2	2	0	0	2	2	2	2	2	1	0	0	0	0	15
P03	2	0	0	0	2	2	2	2	1	1	1	0	0	0	13
P12	1	2	0	0	1	2	1	2	2	3	1	0	0	0	15
P15	0	2	0	0	2	2	2	2	1	2	0	0	0	0	13
P16	3	2	1	2	2	2	2	1	3	2	2	2	1	2	27
P19	0	0	1	1	1	1	2	1	2	1	1	1	1	1	14
P22	3	2	0	0	2	2	2	1	1	3	0	1	1	1	19
P23	2	2	0	2	2	2	2	1	1	2	1	2	0	2	21
P25	2	2	0	2	2	2	2	2	2	3	0	3	0	1	23
P37	2	0	0	0	2	2	2	2	2	2	1	1	0	0	16
P42	2	0	0	2	2	2	2	2	3	1	2	0	0	1	19
P44	2	2	0	0	2	2	2	2	1	3	0	0	0	0	16
P49	2	2	1	2	2	2	2	2	3	1	0	3	1	1	24
P00	0	0	0	0	2	2	2	2	2	1	0	0	0	0	11
P01	0	2	0	2	2	2	2	2	1	3	0	3	0	1	20
P05	2	0	0	0	2	2	2	2	1	1	1	0	0	0	13
P06	0	0	0	0	1	2	1	2	2	1	0	0	0	0	9
P07	0	2	0	0	1	2	1	2	2	1	0	0	0	0	11
P10	0	0	0	0	2	2	0	2	1	1	1	0	1	0	10
P13	1	1	0	1	1	2	1	2	2	2	1	0	0	1	15
P14	0	2	0	0	1	2	1	2	1	2	0	0	0	0	11
P17	1	0	0	0	2	2	0	2	2	2	2	2	0	1	16
P18	3	2	0	0	1	2	1	2	2	2	0	0	0	0	15
P20	0	0	0	0	2	2	2	2	1	1	0	0	0	0	10
P21	3	2	0	0	2	2	2	1	1	3	0	1	1	1	19
P24	0	0	0	0	1	2	1	2	2	1	0	0	0	0	9
P26	2	2	0	2	2	2	2	2	2	3	0	3	0	1	23
P27	3	2	0	0	0	2	1	2	1	3	0	0	0	0	14
P28	2	3	0	0	2	2	2	2	3	3	0	0	0	0	19
P30	1	0	0	1	1	1	1	1	2	1	1	1	0	1	12
P31	0	0	0	1	0	2	1	2	2	3	1	2	0	1	15
P33	1	0	0	1	2	2	0	2	2	3	1	2	0	1	17
P36	2	0	0	1	2	2	2	2	1	3	0	2	0	1	18
P38	2	0	0	0	2	2	2	2	2	2	1	1	0	0	16
P39	2	0	0	0	2	2	2	2	2	2	1	1	0	0	16
P40	2	2	0	0	2	2	0	2	2	3	0	0	0	0	15
Matemática	27	20	0	9	35	45	29	44	39	47	10	18	2	9	334
Informática	23	18	3	11	24	25	25	22	24	25	9	13	4	9	235
Total	50	38	3	20	59	70	54	66	63	72	19	31	6	18	569
Média	1.4	1.1	0.1	0.6	1.6	1.9	1.5	1.8	1.8	2.0	0.5	0.9	0.2	0.5	

APÊNDICE F – Pontuação do *Dr. Scratch*
dos professores e alunos na
etapa de visita  o  s escolas

Programa	Escola	Conceito do PC							Médias		
		Lógica	Paralelismo	Interatividade	Representação dados	Controle de fluxo	Sincronização	Abstração			
https://scratch.mit.edu/projects/313325552	BJ	P22	3	3	2	2	2	3	3	18	16
https://scratch.mit.edu/projects/307085729		P21, P22	1	3	2	2	3	2	1	14	
https://scratch.mit.edu/projects/333407995		a00	1	3	2	1	2	3	1	13	14.3
https://scratch.mit.edu/projects/309984340		a01	1	3	2	2	2	3	1	14	
https://scratch.mit.edu/projects/246359378		a02	1	3	2	2	2	3	3	16	
https://scratch.mit.edu/projects/314092995	DL	P05, P09	2	0	2	1	3	0	1	9	9
https://scratch.mit.edu/projects/312369263		a03	2	1	2	2	3	0	1	11	
https://scratch.mit.edu/projects/312369604		a04	1	0	2	1	3	0	1	8	8.8
https://scratch.mit.edu/projects/312370345		a05	1	0	2	1	3	0	1	8	
https://scratch.mit.edu/projects/312383816		a06	1	2	2	1	1	0	1	8	
https://scratch.mit.edu/projects/312374061		a07	2	0	2	1	3	0	1	9	
https://scratch.mit.edu/projects/312439136		a08	2	0	2	1	3	0	1	9	
https://scratch.mit.edu/projects/324577038		P37	0	0	2	2	1	0	0	5	
https://scratch.mit.edu/projects/324576549	a10	0	0	2	2	1	0	1	6		
https://scratch.mit.edu/projects/324576616	a11	0	1	1	1	1	1	1	6		
https://scratch.mit.edu/projects/324576650	a12	1	2	2	1	2	2	1	11		
https://scratch.mit.edu/projects/324576706	a13	1	1	2	1	2	1	1	9		
https://scratch.mit.edu/projects/324576799	a14	0	2	2	1	2	0	1	8		
https://scratch.mit.edu/projects/324576901	a15	0	1	1	1	1	1	1	6		
https://scratch.mit.edu/projects/324576955	a16	0	1	1	1	1	1	1	6		
https://scratch.mit.edu/projects/324576993	a17	1	1	2	1	2	0	1	8		
https://scratch.mit.edu/projects/324604773	ST	P06, P23	0	0	2	2	1	0	0	5	5.5
https://scratch.mit.edu/projects/356731737		P00, P23	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/324605102		a18	0	0	2	2	1	0	1	6	5.7
https://scratch.mit.edu/projects/324605183		a19	0	0	2	2	1	0	0	5	
https://scratch.mit.edu/projects/324605222		a20	0	0	2	2	1	0	0	5	
https://scratch.mit.edu/projects/324605264		a21	0	1	2	2	1	0	1	7	
https://scratch.mit.edu/projects/324605318		a22	0	0	2	2	1	0	0	5	
https://scratch.mit.edu/projects/324605348		a23	0	0	2	2	1	0	0	5	
https://scratch.mit.edu/projects/324605381		a24	0	0	2	2	1	0	0	5	
https://scratch.mit.edu/projects/324605463		a25	0	0	2	2	1	0	1	6	
https://scratch.mit.edu/projects/324605539		a26	0	0	2	2	1	0	1	6	
https://scratch.mit.edu/projects/324730922		a27	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/324731891		a28	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/324731980		a29	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/324732038		a30	0	0	2	2	1	0	0	5	
https://scratch.mit.edu/projects/324732108		a31	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/324732219		a32	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/324734940		a33	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/324735010		a34	0	0	2	2	2	0	0	6	
https://scratch.mit.edu/projects/317785071		JR	p15, p46	2	1	2	2	2	3	1	
https://scratch.mit.edu/projects/329601915	a35		1	1	2	1	3	1	1	10	
https://scratch.mit.edu/projects/329594075	a36		1	1	2	1	2	0	1	8	10.3
https://scratch.mit.edu/projects/329586222	a37		1	1	2	1	3	0	1	9	
https://scratch.mit.edu/projects/329593745	a38		1	1	2	1	3	1	1	10	
https://scratch.mit.edu/projects/329594365	a39		0	2	2	1	2	1	1	9	
https://scratch.mit.edu/projects/329590188	a40		1	1	2	2	2	1	1	10	
https://scratch.mit.edu/projects/329587841	a41		1	1	2	1	3	0	1	9	
https://scratch.mit.edu/projects/329593643	a42		3	1	2	1	3	1	1	12	
https://scratch.mit.edu/projects/329593798	a43		2	1	2	2	2	0	1	10	
https://scratch.mit.edu/projects/329593867	a44		2	0	2	2	2	1	1	10	
https://scratch.mit.edu/projects/329593679	a45		1	2	2	1	2	0	1	9	
https://scratch.mit.edu/projects/329604755	a46		2	0	2	2	2	1	1	10	
https://scratch.mit.edu/projects/329594528	a47		1	1	2	2	2	0	1	9	
https://scratch.mit.edu/projects/325166729	a48		1	2	2	1	3	0	1	10	
https://scratch.mit.edu/projects/324291021	a49	3	3	2	2	3	3	3	19		
Pontuação total por conceito		Professores	8	7	14	13	14	8	6	Correlação = 0,94	
		Alunos	36	40	95	76	95	25	40		