CEFET/RJ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL POR MEIO DE UMA ABORDAGEM HÍBRIDA: UM ESTUDO DE CASO

Felipe José Rezende de Carvalho

Tese apresentada ao Programa de Pós-Graduação em Ciência Tecnologia e Educação, do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título em Doutor em Ciência, Tecnologia e Educação

Orientador: Marco Antônio Barbosa Braga

Rio de Janeiro
2023

DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL POR MEIO DE UMA ABORDAGEM HÍBRIDA: UM ESTUDO DE CASO

Tese apresentada ao Programa de Pós-Graduação em Ciência, Tecnologia e Educação, do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título de Doutor em Ciência, Tecnologia e Educação.

Felipe José Rezende de Carvalho

Banca examinadora:

_____

Presidente, Professor Doutor Marco Antônio Barbosa Braga (CEFET/RJ)

_____

Professor Doutor Thiago Brañas Melo (CEFET/RJ)

_____

Professor Doutor Tarliz Liao (CEFET/RJ)

_____

Professor Doutor Joel André Ferreira dos Santos (CEFET/RJ)

_____

Professora Doutora Juliana Baptista dos Santos França (UFRJ/RJ)

Rio de Janeiro
Julho de 2023

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

C331 Carvalho, Felipe José Rezende de
Desenvolvimento do pensamento computacional por meio de uma abordagem híbrida: um estudo de caso / Felipe José Rezende de Carvalho. — 2023.
166f. + apêndice : il. color. , enc.

Tese (Doutorado) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, 2023.
Bibliografia : f. 161-166
Orientador: Marco Antônio Barbosa Braga

1. Tecnologia educacional. 2. Inteligência Computacional. 3. Ensino fundamental. 4. Ensino auxiliado por computador. I. Braga, Marco Antônio Barbosa. (Orient.). II. Título.

CDD 371.33

Elaborada pela bibliotecária Tania Mello – CRB/7 nº 5507/04

Ensinar não é transferir conhecimento,
mas criar as possibilidades para a sua
própria produção ou a sua construção.
Paulo Freire

DEDICATÓRIA

Dedico esta tese à minha filha Maria Alice. Durante quase dois anos desse percurso estivemos juntos, em casa, estudando e brincando, bagunçando e arrumando, correndo e pulando. Nos vários momentos de dúvidas que tive sobre conseguir concluir esse trabalho você estava lá, pequenina, com seu sorriso inocente chamando o “papá”, deixando-os mais leves e me fazendo acreditar que era possível. Conseguimos, meu amor. Esse trabalho é para você, papai te ama!

AGRADECIMENTOS

Desenvolver uma tese não é um trabalho solitário. Diversas pessoas se envolvem, participam, contribuem e, por isso, fazem parte dessa conquista, portanto, registro aqui minha gratidão a elas.

Minha esposa, Bruna, companheira fiel que sempre me apoia em todos os meus projetos de vida, sua clareza, maturidade e lucidez são admiráveis. Muito obrigado por fazer essa pesquisa comigo, incentivando, lendo, participando e criticando;

Meus filhos Maria Alice e Miguel, que na inocência de sorrisos e brincadeiras deixam sempre as tarefas árduas mais leves e serenas, obrigado por todo carinho;

Minha família, especialmente meus pais, que sempre me incentivam a avançar, obrigado pelo constante apoio;

Aos colegas do grupo de pesquisa N@melab pelos debates profícuos, especialmente Gustavo, Luis Magro e Fred pelas conversas individuais e auxílios para o avanço desse trabalho, meu muito obrigado;

Luiz d'Escoffier e Afsaneh Hamedi, meus grandes amigos, não tenho palavras para expressar minha gratidão por tudo que fizeram por mim. Nossas infindáveis conversas, leituras e debates sobre esse trabalho contribuíram sobremaneira para eu conseguir estar aqui. Tenho certeza que grande parte dessa conquista é de vocês. Meu eterno agradecimento;

Professor Braga, muito obrigado por confiar em mim e acreditar que eu seria capaz. Nossas conversas com suas orientações sempre instigantes foram essenciais para meu crescimento e conclusão desse trabalho;

Ao grande amigo Léo, muito obrigado! Cara, não sei nem como agradecer por tudo. Suas leituras e críticas sempre sensatas e pertinentes; seu apoio durante a coleta de dados, preparando a sala de informática, copiando os dados. Muito obrigado, meu amigo. Essa vitória também é sua!

Aos professores do programa PPCTE, meus sinceros agradecimentos por todos os ensinamentos passados ao longo desse período;

Aos funcionários da secretaria do programa de pós graduação do CEFET, muito obrigado pelo pronto atendimento às necessidades burocráticas que tive durante o curso;

Aos colegas de departamento do Cap João XXIII, obrigado pelo apoio e substituições nos momentos que precisei estar ausente devido à pós graduação;

Às professoras do Cap João XXIII Anne e Renata, muito obrigado pela gentileza em cederem o espaço de suas aulas para que eu pudesse aplicar a oficina de Pensamento Computacional;

À Universidade Federal de Juiz de Fora, obrigado pelo incentivo à qualificação por meio do programa PROQUALI;

Por fim, não poderia deixar de agradecer a Deus pela minha vida e oportunidades. Sem Ele nada disso seria possível.

RESUMO

O Pensamento Computacional é um estilo de pensamento estruturado que busca resolver problemas das mais diversas áreas. O termo ganhou popularidade a partir da publicação de Wing, em 2006, quando a autora destaca que esse estilo de pensamento, inerente aos cientistas da computação, é tão importante quanto aprender a ler e escrever, devendo ser desenvolvido por todas as pessoas. O destaque de sua relevância no mundo moderno, desde então, influenciou o desenvolvimento de programas e iniciativas educacionais voltadas para a promoção de habilidades de pensamento computacional entre alunos e professores. Ao usar uma combinação de atividades plugadas e desplugadas, os educadores podem fornecer uma experiência de aprendizado de pensamento computacional abrangente, promovendo uma compreensão conceitual mais profunda, e preparando os indivíduos para a solução de problemas em domínios digitais e não digitais. Nossa região de inquérito está nesse âmbito, procurando alternar entre atividades plugadas e desplugadas, objetivando potencializar esse estilo de pensamento. Assim, a questão que guiou nossa pesquisa foi: Como o Pensamento Computacional pode ser potencializado a partir de uma oficina no formato híbrido (mesclando abordagens plugadas, desplugadas e desenvolvimento de projeto)? Para responder à nossa pergunta, projetamos uma oficina de pensamento computacional no formato sugerido, sendo aplicada a três turmas do sétimo ano do Ensino Fundamental II de uma escola pública do interior de Minas Gerais, para posterior análise segundo um estudo qualitativo. Estes dados foram produzidos a partir da aplicação da oficina, e armazenados por meio de vídeos, áudios e registros escritos. Em nossa análise, buscamos observar como os conceitos computacionais - Sequência, Loops, Eventos, Paralelismo, Condicionais, Operadores e Dados estavam sendo manipulados e tangenciados; observamos também como os quatro pilares do PC – Decomposição de Problemas, Abstração, Reconhecimento de Padrões e Algoritmos – foram explorados e manipulados pelos alunos ao longo das atividades. Os dados da pesquisa nos deram pistas sobre a potencialização desse estilo de pensamento nos alunos envolvidos, à medida que avançavam nas atividades, além de ter corroborado a importância da diversidade de abordagens em sala de aula e como os fatores externos atuam fortemente no contexto escolar.

Palavras-chave: Pensamento Computacional; Atividades plugadas e desplugadas; Ensino Fundamental; Programação de computadores; Scratch.

ABSTRACT

Computational Thinking is a style of structured thinking that seeks to solve problems in the most diverse areas. The term gained popularity after the publication of Wing, in 2006, when the author points out that this style of thinking, inherent to computer scientists, is as important as learning to read and write, and should be developed by all people. The highlight of its relevance in the modern world has since influenced the development of educational programs and initiatives aimed at promoting computational thinking skills among students and teachers. By using a combination of plugged in and unplugged activities, educators can provide a comprehensive computational thinking learning experience, fostering deeper conceptual understanding, and preparing individuals for problem solving in both digital and non-digital domains. Our survey region is in this context, seeking to alternate between plugged and unplugged activities, aiming to enhance this style of thinking. Thus, the question that guided our research was: How can Computational Thinking be enhanced from a workshop in a hybrid format (mixing plugged-in, unplugged approaches and project development)? To answer our question, we designed a computational thinking workshop in the suggested format, being applied to three classes of the seventh year of Elementary School II of a public school in the interior of Minas Gerais, for further analysis according to a qualitative study. These data were produced from the workshop application, and stored through videos, audios and written records. In our analysis, we sought to observe how the computational concepts - Sequence, Loops, Events, Parallelism, Conditionals, Operators and Data were being manipulated and tangential; we also observed how the four pillars of the CP – Problem Decomposition, Abstraction, Pattern Recognition and Algorithms – were explored and manipulated by the students throughout the activities. The research data gave us clues about the potentialization of this style of thinking in the students involved as they progressed in the activities, in addition to having corroborated the importance of the diversity of approaches in the classroom and how external factors act strongly in the school context.

Keywords: Computational Thinking; Plugged and unplugged activities; Elementary School; Computer programming; Scratch.

LISTA DE SIGLAS

BNCC	Base Nacional Comum Curricular
CGI	Comitê Gestor da Internet
CSTA	Computer Science Teachers Association
ISTE	International Society for Technology in Education
MEPeCoC	Metodologia para Ensino de Pensamento Computacional para Crianças
PC	Pensamento Computacional
TALE	Termo de Assentimento Livre e Esclarecido
TCLE	Termo de Consentimento Livre e Esclarecido
TDAH	Transtorno do Déficit de Atenção e Hiperatividade
TDIC	Tecnologias Digitais da Informação e Comunicação
TI	Tecnologia da Informação
TIC	Tecnologia da Informação e Comunicação
UFJF	Universidade Federal de Juiz de Fora
UNIOESTE	Universidade Estadual do Oeste do Paraná

LISTA DE FIGURAS

Figura 1 - O ChatGPT 22
Figura 2 - Pilares do Pensamento Computacional - BBC 31
Figura 3 - Pilares do Pensamento Computacional - Brackmann 33
Figura 4 - Nuvem de palavras das definições de Pensamento Computacional 33
Figura 5 - Esquema da MEPeCoc 56
Figura 6 - Cronograma de aula da oficina 57
Figura 7 - Metodologia da pesquisa 62
Figura 8 - Atividade 1 aluno A 66
Figura 9 - Atividade 1 aluno B 67
Figura 10 - Enunciado 1 atividade desplugada 69
Figura 11 - Resolução atividade 1 - A e B 70
Figura 12 - Resolução atividade 1 - C e D 71
Figura 13 - Enunciado atividade 2 desplugada 72
Figura 14 - Resolução atividade 2 - A e B 73
Figura 15 - Resolução atividade 2 C e D 73
Figura 16 - Enunciado atividade 3 desplugada 74
Figura 17 - Resolução atividade 3 - A e B 75
Figura 18 - Resolução atividade 3 - C e D 76
Figura 19 - Enunciado atividade 4 77
Figura 20 - Resolução atividade 4 - A e B 78
Figura 21 - Resolução atividade 4 - C e D 79
Figura 22 - Fase 4 do jogo 80
Figura 23 - Enunciado atividade 1 82
Figura 24 - Resoluções atividade 1 83
Figura 25 - Enunciado atividade 2 83
Figura 26 - Resoluções atividade 2 84
Figura 27 - Enunciado atividade 3 85
Figura 28 - Resolução atividade 3 86
Figura 29 - Enunciado projeto Scratch 87
Figura 30 - Programa Scratch para calcular média de dois valores 90
Figura 31 - Turma 7B fazendo projeto Scratch - 1 92
Figura 32 - Turma 7B fazendo projeto Scratch - 2 92
Figura 33 - Apresentação dos projetos turma 7B 93
Figura 34 - Solução projeto equipe 1 94
Figura 35 - Solução projeto equipe 2 95
Figura 36 - Atividade 1 aluno A 98
Figura 37 - Atividade 1 aluno B 98
Figura 38 - Resolução atividade 1 - A e B 99
Figura 39 - Resolução atividade 1C 100
Figura 40 - Resolução atividade 2 A, B, C 100
Figura 41 - Fase 6 do jogo 101
Figura 42 - Resolução atividade 1 aula 4 103
Figura 43 - Resolução atividade 2 aula 4 103
Figura 44 - Pesquisador conduzindo projeto Scratch 105
Figura 45 - Projeto Scratch aula 6 106
Figura 46 - Projeto Scratch aula 7 107
Figura 47 - Apresentação projeto equipe A 108
Figura 48 - Apresentação projeto equipe B 109

Figura 49 - Resolução atividade 1 113
Figura 50 - Atividade 1 aula 3 115
Figura 51 - Atividade 3.1 aula 4 116
Figura 52 - Atividade 3.2 aula 4 117
Figura 53 - Projeto Scratch aula 6 118
Figura 54 - Projeto Scratch aula 7 119
Figura 55 - Infográfico resumo da oficina de PC 121
Figura 56 - Atividades desplugadas - dia 2, dupla 1 127
Figura 57 - Alunas jogando fase 6 Blockly Games 130
Figura 58 - Alunos construindo labirinto Scratch 131
Figura 59 - Ciclo de ações no contexto da programação de computadores 132
Figura 60 - Atividade 1A aula 4 136
Figura 61 - Atividade 1B aula 4 137
Figura 62 - Programa Scratch – personagem 1 142
Figura 63 - Programa Scratch – personagem 2 143
Figura 64 - Resolução atividade 2 da aula 4 146
Figura 65 - Atividade 1 aluno A 147
Figura 66 - Atividade 1 aluno B 148
Figura 67 - Momento inicial aula 7 149
Figura 68 - Momento final aula 7 150

LISTA DE QUADROS

Quadro 1 - Teses que relacionam Pensamento Computacional e Educação Básica.....42
Quadro 2 - Formato de produção dos dados.....58
Quadro 3 - Resumo das discussões.....151

SUMÁRIO

PREÂMBULO 17
1 INTRODUÇÃO 21
2 QUADRO TEÓRICO 28
2.1 Pensamento Computacional 28
2.2 Formato de avaliação do PC 36
2.3 Construcionismo de Papert 39
2.4 Teses correlatas 42
2.4.1 Texto 1 - Um estudo sobre os estilos de aprendizagem, inteligência lógica e Pensamento Computacional de alunos do Ensino Fundamental e Médio de uma escola colombiana 43
2.4.2 Texto 2 - Desenvolvimento do Pensamento Computacional através de Atividades Desplugadas na Educação Básica 45
2.4.3 Texto 3 - O desenvolvimento do Pensamento Computacional na Educação Matemática segundo contribuições da teoria Histórico-Cultural 45
2.4.4 Texto 4 - O Pensamento Computacional na resolução de problemas na matemática: Uma proposta de aplicação metodológica integrada na Educação Básica... 46
2.4.5 Texto 5 - Pensamento computacional em livros didáticos do ensino médio: sobre atividades e possibilidades 48
2.4.6 Texto 6 - PCOMP-MODEL: Desenvolvendo o Pensamento Computacional na Educação Básica para auxiliar na aprendizagem de algoritmos e programação no Ensino Superior 49
3 PROCEDIMENTOS METODOLÓGICOS 52
3.1 Sobre a postura metodológica 52
3.2 Sobre a produção e coleta dos dados 55
3.3 Sobre os sujeitos da pesquisa 60
3.4 Sobre a forma de discussão e avaliação do PC 60
3.5 Síntese da metodologia de pesquisa 61
4 SOBRE A INTERVENÇÃO 63
4.1 Primeira aplicação da sequência didática 63
4.1.1 7B 63
4.1.1.1 Aula 1 (21/09/2022) – Introdução e atividade desplugada (10:50 às 11:40) 64
4.1.1.2 Aula 2 (22/09/2022) – Atividade desplugada (8:55 às 9:40) 68
4.1.1.3 Aula 3 (22/09/2022) – Atividade plugada (10:50 às 11:40) 80
4.1.1.4 Aula 4 (28/09/2022) – Atividade desplugada (10:50 às 11:40) 82
4.1.1.5 Aula 5 (29/09/2022) – Atividade plugada (8:55 às 9:40) 87

4.1.1.6 Aula 6 (29/09/2022) – Atividade plugada (10:00 às 10:50) 89
4.1.1.7 Aula 7 (30/09/2022) - Atividade plugada (10:00 às 10:50) 91
4.1.1.8 Aula 8 (05/10/2022) – Apresentação final (10:50 às 11:40) 93
4.1.2 7C 97
4.1.2.1 Aula 1 (21/09/2022) - Introdução (10:00 às 10:50) 97
4.1.2.2 Aula 2 (22/09/2022) – Atividade desplugada (10:00 às 10:50) 99
4.1.2.3 Aula 3 (28/09/2022) – Atividade plugada (10:00 às 10:50) 101
4.1.2.4 Aula 4 (29/09/2022) – Atividade desplugada (10:50 às 11:40) 102
4.1.2.5 Aula 5 (30/09/2022) – Atividade plugada (8:55 às 9:40) 104
4.1.2.6 Aula 6 (30/09/2022) – Atividade plugada (10:50 às 11:40) 105
4.1.2.7 Aula 7 (05/10/2022) – Atividade plugada (8:55 às 9:40) 106
4.1.2.8 Aula 8 (05/10/2022) – Apresentação final (10:00 às 10:50) 107
4.2 Segunda aplicação da sequência didática 111
4.2.1 7A 111
4.2.1.1 Aula 1 (19/10/2022) - Introdução (10:00 às 10:50) 111
4.2.1.2 Aula 2 (19/10/2022) – Atividade desplugada (10:50 às 11:40) 113
4.2.1.3 Aula 3 (21/10/2022) – Atividade plugada (7:20 às 8:05) 114
4.2.1.4 Aula 4 (21/10/2022) – Atividade desplugada (8:05 às 8:55) 116
4.2.1.5 Aula 5 (26/10/2022) – Atividade plugada (10:00 às 10:50) 117
4.2.1.6 Aula 6 (26/10/2022) – Atividade plugada (10:50 às 11:40) 118
4.2.1.7 Aula 7 (28/10/2022) – Atividade plugada (07:20 às 08:05) 118
4.2.1.8 Aula 8 (28/10/2022) – Apresentação final (08:05 às 8:55) 119
4.3 Síntese das aplicações 120
5 ANÁLISE E DISCUSSÃO DOS DADOS 122
5.1 Aula 1 123
5.2 Aula 2 125
5.3 Aula 3 129
5.4 Aula 4 135
5.5 Aulas 5, 6 e 7 139
5.6 Casos que emergiram da análise 144
5.6.1 O caso Joaquim 144
5.6.2 O caso TDAH 146
5.7 Uma síntese em relação a análise 151
6 CONSIDERAÇÕES FINAIS 155
REFERÊNCIAS 161
APÊNDICE A – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS - AULA 2 167

APÊNDICE B – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS – AULA 4 PARTE INICIAL 169
APÊNDICE C – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS – AULA 4 PARTE FINAL - 1ª APLICAÇÃO 170
APÊNDICE D – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS – AULA 4 PARTE FINAL - 2ª APLICAÇÃO 171
APÊNDICE E – DISCUSSÃO INICIAL PARA OS PROJETOS SCRATCH 173
APÊNDICE F – PROBLEMAS COVID PARA RESOLVER NO SCRATCH 174
APÊNDICE G – ROTEIRO DA ENTREVISTA COLETIVA 179
APÊNDICE H – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO 180
APÊNDICE I – TERMO DE ASSENTIMENTO LIVRE E ESCLARECIDO 182

PREÂMBULO

Como forma de deixar claro ao leitor o percurso do pesquisador até culminar na presente pesquisa, este capítulo introdutório está dividido em duas seções, sendo a primeira, escrita em primeira pessoa, um percurso pessoal do pesquisador até o momento atual, e a segunda, uma introdução ao presente estudo.

Optamos por apresentar esse percurso pessoal buscando apresentar ao leitor a região de fala do pesquisador, como o trabalho com o Pensamento Computacional perpassa a vida do pesquisador desde sua formação inicial.

Percurso pessoal

No ano de 2005, iniciei minha licenciatura plena em Matemática pela Universidade Presidente Antônio Carlos, na cidade de Barbacena, interior de Minas Gerais. Naquele momento, muito jovem, tendo concluído recentemente o Ensino Médio, ainda não sabia exatamente qual caminho profissional trilhar, porém, influenciado pelos pais professores de Matemática, enveredei pelo mesmo caminho.

Paralelamente a essa graduação, em 2006, ingressei no curso de Tecnologia em Sistemas para a Internet, no Instituto Federal da mesma cidade e, de maneira concomitante, cursava as duas graduações, tendo concluído a primeira em meados de 2008 e a segunda no final de 2010. Faço esse destaque inicial de minha formação dupla, pois sempre gostei do trabalho com as tecnologias digitais, motivo pelo qual fui atraído a cursar a segunda graduação.

Minha história como educador teve início em março de 2009, primeiro momento em que assumi uma classe lecionando estatística básica para um curso técnico em Segurança do Trabalho. Após momentos de ansiedade e incerteza, a paixão pela sala de aula foi tomando conta e estava cada vez mais envolvido nesse contexto. Ainda nessa mesma instituição, passei a lecionar informática instrumental para vários cursos e logo o contexto escolar se tornou familiar para mim.

Com o passar do tempo, ganhando experiência nas várias instituições que lecionei, fui me incomodando com o formato de aula que trabalhava. Naquele momento, aulas ditas tradicionais, da forma como fui ensinado e formado, exposições na lousa e listas de exercícios. A vontade de tentar fazer algo diferente, utilizar os conhecimentos de informática para contribuir com as aulas de matemática, me motivou a fazer uma especialização lato sensu em

Educação Matemática Comparada, cursada na Escola Superior Aberta do Brasil (ESAB) na modalidade EAD, tendo concluído o curso em 2013, apresentando um trabalho de conclusão intitulado *Evolução da educação: a informática nas aulas de matemática.* Esse trabalho de conclusão de curso trazia, de maneira geral, pequenas reflexões sobre utilização da informática para contribuir no ensino da matemática, utilização de planilha eletrônica para trabalhar números decimais, funções, construções de gráficos, utilizar o superlogo para construir formas geométrica e um jogo de labirinto para explorar aritmética básica. Mas foi em 2014 que tive o maior salto intelectual, ao cursar uma disciplina de mestrado como ouvinte na Universidade Federal de Ouro Preto.

A disciplina Ambientes Educacionais Informatizados, lecionada pela professora Regina Franchi, abordou debates teóricos sobre o assunto e, na parte prática, conheci o *software* Scratch[1] que permite, de forma lúdica e com linguagem de fácil acesso, a programação de computadores, possibilitando a construção de histórias e animações. Foi ali que conheci a possibilidade de construir na sala de aula um ambiente no qual o aluno tenha mais protagonismo frente às tecnologias digitais, desenvolva trabalhos e projetos que o permita avançar no desenvolvimento de seu conhecimento.

No ano seguinte, 2015, ingressei através de concurso público no Instituto Federal do Paraná, Campus Capanema. No interior do Paraná, comecei a lecionar Matemática para alunos do Ensino Médio desse Instituto Federal e, nesse ambiente de trabalho, tive a oportunidade de desenvolver um projeto com os alunos sobre a construção de conhecimentos de matemática por meio do desenvolvimento de jogos no *software* Scratch. Na construção desses jogos os alunos tinham a oportunidade de explorar conceitos matemáticos em um formato mais lúdico, além de se familiarizarem com a programação de computadores por meio da programação visual. Alguns dos frutos desse trabalho foram apresentados no Encontro Nacional de Educação Matemática, estando disponíveis em Carvalho (2016).

Minha trajetória na pesquisa acadêmica tem início em fevereiro de 2016, com o ingresso no Mestrado Acadêmico em Ensino, da Universidade Estadual do Oeste do Paraná (UNIOESTE), em Foz do Iguaçu. Nesse curso, pude amadurecer um pouco sobre a pesquisa no contexto educacional e avançar mais sobre o trabalho com as tecnologias digitais na educação, desenvolvendo uma pesquisa sobre a resolução de uma tarefa de Modelagem Matemática no contexto de programação de computadores com Scratch, favorecendo o desenvolvimento do Pensamento Computacional.

[1]https://scratch.mit.edu/

Essa pesquisa foi desenvolvida em uma escola no interior do estado de Minas Gerais, pois, durante o mestrado, prestei um novo concurso público para trabalhar no Colégio de Aplicação da Universidade Federal de Juiz de Fora e, sendo aprovado, retornei a Minas Gerais como docente de Matemática do Ensino Fundamental e Médio dessa escola. Foi nesse colégio que dei sequência à pesquisa de mestrado e, em um projeto no contraturno das aulas, coletei os dados.

O enfoque da pesquisa foi uma tarefa de modelagem matemática, sendo nela discutido com os alunos do primeiro ano do Ensino Médio o problema do trânsito. A partir de reportagens jornalísticas, problematizamos cruzamentos da cidade de Juiz de Fora – MG que não dispunham de semáforos. Os alunos precisaram modelar a implementação de semáforos auxiliados pelo ambiente de programação Scratch, preocupando-se com sincronia desses semáforos, além do tempo das luzes para travessia de pedestres. Mais detalhes dessa pesquisa podem ser acessados em Carvalho (2018).

Quando retornei a Minas Gerais em meados de 2016 para trabalhar no Cap João XXIII da UFJF, conheci um pouco da Robótica Educacional tornando-a parte de meu trabalho a partir de 2017. Inicialmente um trabalho mais lúdico e de admiração com as suas possibilidades, que depois se tornou alvo de reflexões: o que é possível ensinar com a robótica? O simples manuseio de equipamentos robóticos e a montagem de pequenos artefatos proporcionam algum tipo de aprendizado? Quais conhecimentos pedagógicos são explorados no trabalho com a robótica? Quais competências são mobilizadas nesse tipo de trabalho? Essas questões foram surgindo à medida que avançava no trabalho com a robótica.

Paralelamente a esse trabalho com a robótica, e também de forma complementar a ela, alguns projetos envolvendo a programação de computadores, especialmente preconizando pela programação em blocos com o *software* Scratch, foram desenvolvidos com alunos do Ensino Fundamental I, além de projetos com alunos do Ensino Médio.

Esses trabalhos voltados a explorar as tecnologias digitais de diversos modos ocorreram no âmbito do Colégio de Aplicação como parte de meu trabalho docente. Nesse ambiente foi possível trabalhar também em parceria com alunos da Engenharia da Universidade Federal de Juiz de Fora por meio do programa de treinamento profissional (TP), além das bolsas de iniciação científica (BIC Jr). Os alunos da engenharia participaram ativamente dos projetos. No projeto de programação para crianças do 4º ano do Ensino Fundamental, ajudavam na preparação das aulas e também durante a execução, auxiliando as crianças a construírem seus projetos no Scratch; no projeto de robótica educacional, participavam dos encontros semanais

do grupo ajudando a desenvolver projetos com os equipamentos de robótica além de sua programação.

Um pouco do que foi desenvolvido ao longo desses anos está no blog[2] que construímos como forma de registrar e divulgar nosso trabalho, podendo servir de auxílio a outros profissionais que desejem desenvolver esse tipo de atividade. Já em Silva e Carvalho (2018), são apresentadas algumas reflexões sobre um projeto de robótica desenvolvido com estudantes do Ensino Médio.

Todos esses trabalhos e projetos com as Tecnologias Digitais permearam, e ainda permeiam, minha trajetória profissional. Nesse viés, inquietações sobre o desenvolvimento do Pensamento Computacional nesse meio surgem. Mais recentemente, trabalhos com o intuito de explorar esse tipo de pensamento sem a utilização das tecnologias digitais (de forma desplugada) estão sendo desenvolvidos. Olhando para esses trabalhos, algumas inquietações surgem: Como podemos explorar esse tipo de pensamento nesse formato desplugado? E um trabalho de forma híbrida, utilizando-se de abordagens com e sem dispositivos digitais?

Na presente, pesquisa pretendo adensar discussões em relação a essas questões a partir de alguns trabalhos desenvolvidos com alunos da Educação Básica (Ensino Fundamental) que objetivaram explorar o Pensamento Computacional utilizando uma abordagem que mesclou atividades plugadas e desplugadas.

[2] O blog apresenta algumas atividades desenvolvidas com robótica e também com a programação de computadores para as crianças. Disponível em < https://roboticajoaoxxiii.wixsite.com/ufjf >. Acesso em: 28 de mar. de 2023.

1 INTRODUÇÃO

As tecnologias digitais estão cada vez mais imbricadas em nossa sociedade. Os artefatos tecnológicos estão presentes em nossas interações sociais, nos momentos de lazer, trabalho, entre outras. Dificilmente conseguimos imaginar algumas de nossas atribuições do dia a dia que sejam isentas do uso desses equipamentos. Castells (2005) afirmou que, a partir da década de 1960, um novo paradigma tecnológico baseado nas tecnologias de informação e comunicação começava a tomar forma.

Hoje, não há dúvidas que vivemos em um mundo conectado. León (2021) aponta que o Brasil possui mais de 150 milhões de pessoas com acesso à internet, mesmo sendo esse acesso ainda desigual. Esse valor é corroborado pelo resumo executivo do Comitê Gestor da Internet (CGI)[3], que destaca que em 2022, 82% dos domicílios brasileiros possuíam acesso à internet, tendo esse valor apresentado uma crescente, mesmo nas residências rurais. Há de se destacar, todavia, que em grande medida somos apenas consumidores de informações e tecnologias. Mudar essa lógica de consumidores para construtores de tecnologia tem sido um desafio a que diversos educadores têm se proposto.

Trabalhar o caráter construtor de tecnologias desde a Educação Básica pode ser, por exemplo, uma forma de tentar minimizar a defasagem que há de profissionais na área de Tecnologia da Informação (TI) no Brasil. Em uma reportagem veiculada pelo jornal da Universidade de São Paulo, Ferro (2021) destaca essa defasagem. Conforme apontado, o processo de automação que estamos vivenciando está demasiadamente acelerado, e não há formação de mão de obra suficiente para suprir essa necessidade.

A Associação das Empresas de Tecnologia da Informação e Comunicação (TIC) e de Tecnologias Digitais (BRASSCOM), fez um estudo[4] corroborando essa defasagem. Conforme a associação, haverá uma demanda de 797 mil profissionais da área de tecnologia até o ano de 2025, sendo que a formação desses profissionais não consegue acompanhar tal demanda.

Essa defasagem de profissionais na área de Tecnologia da Informação nos permite depreender que, talvez, haja desinteresse ou desconhecimento dos jovens por essa carreira, ou por julgarem-na como uma carreira difícil. Talvez a inserção do Pensamento Computacional

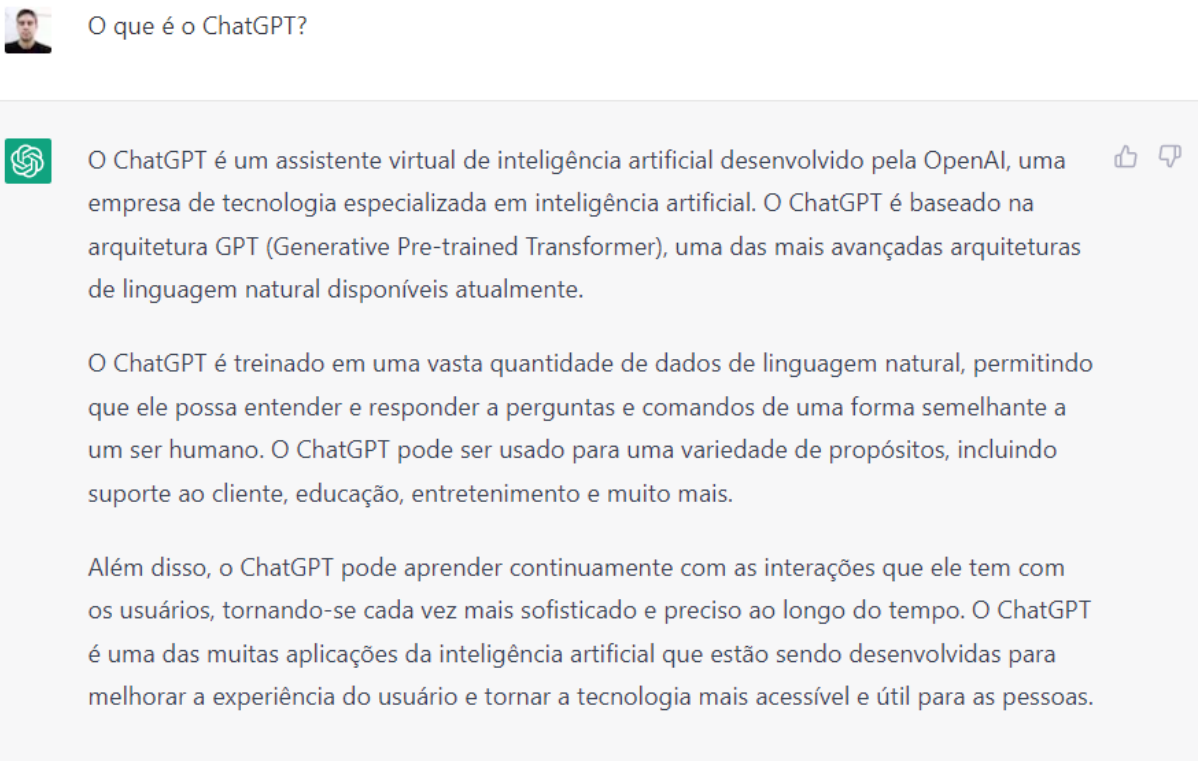
[3] Relatório completo disponível em <https://cetic.br/pt/tics/domicilios/2022/domicilios/A4/>. Acesso em 12 de jun. de 2023.
[4] Disponível em <https://brasscom.org.br/estudo-da-brasscom-aponta-demanda-de-797-mil-profissionais-de-tecnologia-ate-2025/>. Acesso em 2 de fev. de 2023.

(PC) desde a Educação Básica, a partir de atividades lúdicas e projetos que sejam do interesse dos alunos, também possa fazer com que construam uma nova visão a respeito.

Os avanços tecnológicos e as ferramentas cada vez mais modernas reforçam a necessidade de desenvolvimento do PC e da capacidade de resolução de problemas pelas pessoas. As tecnologias mais avançadas que utilizam inteligência artificial estão cada vez mais próximas de nós. Recentemente, a empresa OpenAI[5] lançou uma ferramenta interativa no formato de chat. Essa ferramenta, que por enquanto é gratuita, utiliza inteligência artificial para conseguir "conversar" com o usuário como se fosse uma pessoa, buscando suas informações na internet. Conforme explicado no site da empresa, quanto mais o chat é utilizado, mais ele aprende com os humanos.

Como curiosidade, perguntamos ao próprio ChatGPT o que ele era. A Figura *1* apresenta o resultado.

Figura 1 - O ChatGPT



Fonte: o autor

Há uma infinidade de possibilidades para a utilização dessa tecnologia. No contexto escolar, há uma grande preocupação por parte de alguns professores, visto que essa tecnologia

[5] É uma empresa de pesquisa e implantação de Inteligência Artificial com sede na Califórnia, EUA. Disponível em <https://openai.com/>. Acesso em 23 de mar. de 2023

é capaz de “escrever” redações, pesquisar sobre qualquer assunto, resolver questões de matemática etc. Especialistas afirmam[6] ser uma perda de tempo tentar proibir a utilização dessa tecnologia no contexto escolar, e discutem formas de como utilizá-la a favor do ensino, sem que seja apenas um canal para fazer cópias, sem criticidade por parte dos alunos.

A verdade é que, diante de tantas possibilidades que se abrem com tais tecnologias, o contexto escolar não pode ficar alheio a isso, e o trabalho em prol do desenvolvimento do PC pode ser um caminho no sentido de proporcionar aos alunos um avanço em relação às tecnologias para além de simples consumidores.

Na década de 1960, o educador Seymour Papert já apresentava essas preocupações em relação a construção de conhecimento pelos alunos a partir do trabalho com programação de computadores. Naquela época, ele desenvolveu a linguagem de programação de computadores LOGO, almejando proporcionar meios para que as crianças fossem construtoras de tecnologias, e não somente consumidoras (PAPERT, 1985). Basicamente, essa linguagem consistia em "ensinar" uma tartaruga virtual, por meio de códigos de programação, a se movimentar, construindo, por exemplo, formas geométricas.

A ideia preconizada por Papert é que houvesse uma mudança do instrucionismo[7] para o construcionismo, ou seja, para o autor, "a meta é ensinar de forma a produzir a maior aprendizagem a partir do mínimo de ensino" (PAPERT, 1994, p. 125). As ideias de Papert serão trazidas com maior profundidade em uma seção específica dentro do quadro teórico da tese.

Resnick (2012), rememorando as ideias de Papert, destaca que o sonho idealizado por ele era transformar o computador em um meio de construção de conhecimento pelas crianças, porém, conforme ressalta, isso ainda é apenas sonho. Há algumas iniciativas nesse sentido, mas ainda tímidas. Resnick (2012) destaca que os computadores foram se tornando, no meio escolar, apenas mais uma forma de transmitir informações. O potencial de permitir a construção de conhecimentos foi mitigando, e os formatos tradicionais de ensino foram sendo digitalizados. Conforme o autor, apesar das crianças serem consideradas nativas digitais por terem nascido em um momento em que essas tecnologias estão bastante difundidas, elas são apenas consumidoras desses recursos. Para Resnick (2012), apenas consumir essas tecnologias é o mesmo que ler e não saber escrever.

---

[6] Professores da área de tecnologia que fazem essa afirmação. Mais informações disponíveis em <https://g1.globo.com/educacao/noticia/2023/01/29/tentar-proibir-chatgpt-nas-escolas-sera-perda-de-tempo-dizem-especialistas-veja-pros-e-contras-do-robo-na-sala-de-aula.ghtml>. Acesso em 23 de mar. de 2023

[7] Segundo Papert (1994), a filosofia instrucionista prega que, quanto melhor a instrução, melhor o aluno aprende. Essa filosofia faz oposição ao construcionismo que, segundo o autor, o aluno constrói seu próprio conhecimento. Mais detalhes sobre essas filosofias serão apresentadas no terceiro capítulo da tese.

A valorização do aluno enquanto protagonista[8] na resolução de problemas de geometria por meio da programação na linguagem LOGO, proposta por Papert, pode ser considerada como os primórdios do que hoje se conhece como o Pensamento Computacional. Esse termo ganhou popularidade a partir da publicação da cientista da computação norte americana Jeannette Wing, no ano de 2006. Em seu artigo intitulado "Computational Thinking", a autora ressalta a importância de se pensar como um cientista da computação, destacando que esse tipo de pensamento está ligado à formulação e resolução de problemas.

Em sua publicação, Wing (2006) aponta que o PC pode ser desenvolvido por meio da programação de computadores, porém, não está limitado a ela, podendo inclusive ser desenvolvido sem a utilização de equipamentos digitais. Tal fato é importante de se destacar visto que, antes dessa publicação, apesar do termo não ser difundido, o seu aprendizado era tido como consequência da programação de computadores e, após a publicação de Wing, ele ganha um significado mais amplo e complexo, sendo a programação de computadores mais uma possibilidade a partir do seu desenvolvimento (GUARDA; PINTO, 2020).

Trabalhar o Pensamento Computacional na Educação Básica tem sido defendido por diversos autores (WING, 2006; BUNDY, 2007; BRACKMANN, 2017), sendo ressaltado seu potencial para resolução de problemas nas mais diversas áreas. Esse tipo de pensamento favorece o poder de abstração a partir de um trabalho interdisciplinar, sendo importante não só para os cientistas da computação, mas para qualquer pessoa. Wing (2006; 2017) ressalta que além das habilidades básicas de ler, escrever e calcular, deveria ser também acrescido o PC à capacidade analítica de toda criança.

O Pensamento Computacional hoje é explorado no contexto escolar principalmente por meio da programação de computadores, mas também há abordagens que envolvem robótica e jogos digitais (AVILA, CAVALHEIRO, 2017; FERREIRA et al, 2019; PACIERI, SIQUEIRA, OLIVEIRA, 2019; SOUZA, SAMPAIO, ANDRADE, 2018; SILVA et al, 2016).

Os trabalhos com robótica no contexto escolar podem incluir desde a construção de artefatos até a transposição de desafios a partir da programação desses equipamentos. Na atividade de robótica objetivando desenvolver o PC, apresentada por Zanetti e Oliveira (2015), por exemplo, os alunos precisavam programar um carrinho robótico para contornar um obstáculo.

[8] Utilizamos o termo “protagonista” nesta tese aludindo ao que Papert preconizou ao desenvolver a linguagem de programação LOGO. A ideia é que o aluno, ser cognoscente, seja capaz de “ensinar” o computador, e não o contrário. Com isso, não estamos dizendo que o papel do professor fique minimizado no contexto escolar, mas queremos reforçar a importância do aluno produzir durante as aulas.

Ao contrário do que a expressão sugere, o PC também pode ser trabalhado sem a utilização de qualquer equipamento tecnológico, de maneira desplugada. A programação desplugada explora jogos e atividades criativas que podem proporcionar o desenvolvimento desse estilo de pensamento. Utilizando um tabuleiro de xadrez, por exemplo, lateralidade, resolução de problemas e planejamento de rotas podem ser explorados (MARISTALAB, 2021). Em sua pesquisa sobre o desenvolvimento do PC utilizando abordagens desplugadas, Brackmann (2017) apresenta uma série de jogos de tabuleiro e desafios no papel que foram utilizados em seu trabalho.

No Brasil, o Pensamento Computacional está ganhando mais relevância, e abordagens têm sido cada vez mais desenvolvidas, porém, ainda não há uma política pública para inserir o PC como uma disciplina escolar. Na Base Nacional Comum Curricular (BNCC) o termo é citado relacionando-o com a matemática, especialmente no que diz respeito a investigação e resolução de problemas. Ao relacionar o PC com a matemática, a BNCC ressalta que

> Associado ao pensamento computacional, cumpre salientar a importância dos algoritmos e de seus fluxogramas, que podem ser objetos de estudo nas aulas de Matemática. Um algoritmo é uma sequência finita de procedimentos que permite resolver um determinado problema. Assim, o algoritmo é a decomposição de um procedimento complexo em suas partes mais simples, relacionando-as e ordenando-as, e pode ser representado graficamente por um fluxograma. A linguagem algorítmica tem pontos em comum com a linguagem algébrica, sobretudo em relação ao conceito de variável. Outra habilidade relativa à álgebra que mantém estreita relação com o pensamento computacional é a identificação de padrões para se estabelecer generalizações, propriedades e algoritmos (BRASIL, 2018, p.271).

Navarro (2021), destacando a estreita relação do PC com a matemática, afirma que ele está especialmente ligado à resolução de problemas, ao pensamento algébrico e algorítmico. Apesar dessa relação mais estreita com a matemática, abordagens associadas a outras disciplinas também são possíveis, especialmente no que diz respeito à resolução de problemas. Charão *et al*. (2018), por exemplo, apresentam um esforço de trabalho interdisciplinar entre PC e psicologia.

A partir do exposto, observamos a importância e relevância do tema ser trabalhado desde a Educação Básica, possibilitando o avanço intelectual e o desenvolvimento de um estilo de pensamento que contribui para a resolução de problemas. Assim sendo, apontamos que nossa região de inquérito está no trabalho articulado do Pensamento Computacional plugado e desplugado, com o desenvolvimento de pequenos projetos em um *software* de programação visual. A presente pesquisa procura olhar para uma sequência de atividades que tangencia nossa região de inquérito.

A tese que ora nos propomos a desenvolver é a de que, a introdução ao PC por meio dessa abordagem híbrida, articulado ao desenvolvimento de projetos de programação, pode potencializar a construção/aperfeiçoamento desse tipo de pensamento. Segundo o dicionário Priberam[9] da língua portuguesa, a palavra "potencializar" tem como significado "tornar potente, reforçar". Em nosso estudo, utilizaremos a palavra conforme a segunda definição, entendendo que o PC, em alguma medida, já faz parte do indivíduo, podendo, ao longo da vida, ser desenvolvido, despertado, intensificado, adaptado, portanto, potencializado.

Conforme apontado pela literatura, é possível explorar o PC sem a utilização de recursos tecnológicos digitais, porém, não há dúvidas de que as tecnologias digitais permitem explorações e resoluções que não são possíveis por outros meios. Assim, entendemos que associar essas duas abordagens com projetos de programação pode ser promissor ao desenvolvimento do PC.

A partir dos apontamentos gerais da literatura que fizemos, observamos haver consenso sobre a importância de se desenvolver o PC desde a Educação Básica, porém, não há um único meio para fazê-lo. Assim, a partir dessa problemática, mergulhamos em nossa pesquisa buscando apresentar uma nova possibilidade para o trabalho com o PC, bem como uma definição e análise para esse estilo de pensamento. Para avançar em nossa pesquisa, procuramos sintetizar a inquietação por meio da questão: Como o Pensamento Computacional pode ser potencializado a partir de uma oficina no formato híbrido (mesclando abordagens plugadas, desplugadas e desenvolvimento de projeto)?

O objetivo geral da pesquisa é compreender como o Pensamento Computacional pode ser desenvolvido, adaptado, potencializado por meio da oficina que mescla atividades plugadas e desplugadas.

Os objetivos específicos são:

- Analisar como cada atividade, plugada e desplugada, pode contribuir, ou não, para a potencialização do PC;

- Compreender de que modo a articulação entre atividades plugadas e desplugadas podem favorecer a potencialização do PC;

- Estudar como o formato híbrido da oficina pode contribuir para repensar práticas educacionais;

- Determinar que outros elementos, para além do PC, o formato híbrido da oficina pode fomentar.

[9] Disponível em <https://dicionario.priberam.org/potencializar>. Acesso em 13 de mar. 2023

No intuito de buscar elementos para compreender nossa inquietação e tangenciar o objetivo dessa pesquisa, construímos uma oficina para trabalhar o PC. Essa oficina teve como objetivos:

- proporcionar experiências plugadas e desplugadas que explorem conceitos do PC;
- explorar conceitos relacionados ao PC por meio da programação em blocos.

Os detalhes em relação a essa oficina serão expostos no terceiro capítulo da tese, referente à metodologia. Esse capítulo ainda detalhará a postura metodológica adotada, a forma utilizada para construção, coleta e análise dos dados, e ainda discorrerá sobre os sujeitos da pesquisa.

O segundo capítulo da tese apresentará o quadro teórico que embasou nosso trabalho. Nele, traremos discussões em relação ao Pensamento Computacional e sua avaliação, além de apresentar a teoria construcionista de Seymour Papert e algumas teses correlatas.

O capítulo quatro da tese traz uma descrição minuciosa e detalhada da intervenção. Esse capítulo apresenta cada uma das aulas, com ilustrações das atividades desenvolvidas pelos alunos, bem como algumas falas. Esse capítulo serviu de base para a discussão e análise dos dados, que é apresentada no capítulo cinco da tese.

O sexto e último capítulo apresenta as considerações finais do trabalho. Nesse capítulo procuramos retomar a questão de pesquisa bem como os objetivos, traçando considerações a partir da análise dos dados.

Após as considerações finais, são apresentadas as referências utilizadas ao longo da tese e, em seguida, nos apêndices, todos os enunciados das atividades foram disponibilizados, bem como o roteiro da entrevista coletiva e os termos de consentimento que foram assinados pelos alunos e seus responsáveis.

2 QUADRO TEÓRICO

No presente capítulo procuramos apresentar um pouco da literatura que permeia o Pensamento Computacional e sua avaliação, bem como a teoria construcionista. Ao final do capítulo é apresentado algumas teses com a temática correlata a nossa. O aprofundamento e discussão dessa literatura se mostrou fundamental para que pudéssemos avançar na compreensão de como o PC pode ser potencializado a partir de uma oficina no formato híbrido.

2.1 Pensamento Computacional

No ano de 2006, a professora Jeannette Wing da Universidade de Columbia publicou um artigo de opinião sobre o Pensamento Computacional. Em sua publicação, ela afirma que o PC é tão importante quanto leitura, escrita e matemática básica. Para ela, esse pensamento é uma competência fundamental para qualquer pessoa, não apenas para os profissionais da computação (WING, 2006). Em suas palavras, "o pensamento computacional envolve resolver problemas, projetar sistemas e entender o comportamento humano, baseando-se nos conceitos fundamentais da ciência da computação" (WING, 2006, p.33 – tradução nossa).

Ainda nesse mesmo artigo, Wing (2006) destaca que esse pensamento não é desenvolvido simplesmente por meio dos computadores, destacando seu caráter mais amplo e complexo, sendo composto por uma gama de ferramentas mentais que auxiliam na resolução de problemas, projeção de sistemas e entendimento do comportamento do ser humano. Ela destaca que há uma confusão entre o PC e programação de computadores. Na verdade, o PC pode ser desenvolvido pela programação, porém, não está limitado a ela.

A partir da publicação de Wing, em 2006, o termo ganha popularidade e diversas publicações e pesquisas começam a ser desenvolvidas buscando explorar e entender melhor esse tipo de pensamento. Em 2007, Bundy afirmou que esse tipo de pensamento está permeando quase todas as ciências. Conforme esse autor, a lógica do PC tem influenciado diversas outras formas de pensamento, além das ciências naturais e tecnologias, perpassando também pelas ciências sociais. Conforme o autor, o entendimento do século XXI está vinculado aos fundamentos da computação.

Ainda nesse viés interdisciplinar, Nunes (2011) defende a pluralidade de aplicação do pensamento computacional desde a Educação Básica. Conforme aponta, há um avanço no processo cognitivo que temos desenvolvido ao longo das últimas décadas, sendo rotineira a utilização de diferentes algoritmos na solução de problemas das mais diversas áreas.

Apesar de muito se falar no termo, uma definição exata do que seja ainda não há. Em relação a essa falta de definição, Hemmendinger (2010) tece críticas. Para ele, Wing reuniu diversas características da racionalização utilizada na resolução de problemas, como a lógica matemática, as diferentes formas de metodologias científicas e uso da modelagem naquilo que denominou pensamento computacional. Nessa mesma linha, Hu (2011) afirma que esse tipo de pensamento pode ser um híbrido da capacidade que as pessoas adquirem a partir de diversos meios. Mas, será mesmo necessário ter uma definição exata do que seja o PC?

Conforme Hu (2011) aponta, talvez o mais importante seja desenvolver uma cultura computacional no contexto escolar, ou, conforme Hemmendinger (2010), o mais importante seja ensinar a utilizar o computador para auxiliar na resolução de problemas de diversas áreas, criando e descobrindo novas soluções. Conforme ele ainda aponta, a ciência da computação pode ajudar, mas não será a solução de todos os problemas.

No ano de 2011, a International Society for Technology in Education (ISTE), em conjunto com a Computer Science Teachers Association (CSTA), divulgaram um documento com uma definição operacional para o PC. Essa definição teve o aval de quase 700 professores da área (ISTE; CSTA, 2011). Assim, definiram que:

> O Pensamento Computacional é um processo de resolução de problemas que inclui (mas não está limitado a) as seguintes características:
> - Formulação de problemas de forma que nos permita usar um computador e outras ferramentas para nos ajudar a resolvê-los;
> - Organização e análise lógica de dados;
> - Representação de dados através de abstrações, como modelos e simulações;
> - Automatização de soluções através do pensamento algorítmico (uma série de etapas ordenadas);
> - Identificação, análise e implementação de possíveis soluções com o objetivo de alcançar a combinação mais eficiente e efetiva de etapas e recursos;
> - Generalização e transferência de um determinado processo de resolução de problemas, para uma grande variedade de problemas.

Sobre o documento divulgado pela ISTE e CSTA (2011), Barr e Stephenson (2011) afirmam que ele se trata de um esforço para tentar entender a abrangência do PC para além da ciência da computação, ressaltando que uma definição operacional para o PC deve se preocupar especialmente com sua aplicação em sala de aula.

Brennam e Resnick (2012, p.3) desenvolveram uma definição para o Pensamento Computacional que procura articular três dimensões:

> Conceitos computacionais (os conceitos que os designers empregam enquanto programam), práticas computacionais (as práticas que os designers desenvolvem enquanto programam) e perspectivas computacionais (as perspectivas que os designers formam sobre o mundo ao seu redor e sobre si mesmos) (tradução nossa)

A definição construída pelos autores se deu a partir de alguns anos de observação dos projetos Scratch desenvolvidos por crianças, além de entrevistas com esses desenvolvedores. Assim, os autores destacam que o Pensamento Computacional é a junção de conceitos computacionais com práticas e perspectivas. Conforme apontam, há 7 conceitos computacionais que permeiam o PC, quais sejam:

1) Sequências: uma sequência de instruções especifica o comportamento ou ação a ser produzida;
2) Loops: mecanismo para executar mesma sequência diversas vezes;
3) Eventos: uma coisa faz com que outra aconteça;
4) Paralelismo: sequência de instruções que acontecem ao mesmo tempo;
5) Condicionais: capacidade de tomar decisões com base em certas condições;
6) Operadores: oferecem suporte para expressões matemáticas, lógicas e de *strings*;
7) Dados: envolvem armazenar, recuperar e atualizar valores;

Além desses conceitos, Brennam e Resnick (2012) destacam as 4 práticas que se concentram no processo de pensar e aprender, indo além do que se aprende e tangenciando o como se aprende.

1) Ser incremental e iterativo: um projeto se torna adaptativo de acordo com seu andamento;
2) Teste e depuração: um projeto raramente funciona exatamente como planejado. Ao longo do desenvolvimento acontecem testes e ajustes do projeto.
3) Reutilizar e remixar: construir seu trabalho a partir do trabalho dos outros permite desenvolvimento da capacidade de leitura crítica dos códigos;
4) Abstrair e modularizar: capacidade de construir algo grande a partir da junção de coleções de peças menores.

De acordo com os autores, há ainda 3 perspectivas que se somam aos conceitos e práticas para formar um pensador computacional.

1) Expressar-se: um pensador computacional vê a computação como mais do que algo a construir, mas algo que o permite expressar-se, mostrar suas ideias;
2) Conexão: aprendizagem e criatividade são práticas sociais; projetar mídias com Scratch é enriquecido pelas interações com os outros;
3) Questionamento: jovens devem se sentir capacitados a fazerem perguntas com e sobre as tecnologias.

É importante relembrar que toda essa articulação sobre o Pensamento Computacional que Brennam e Resnick (2012) fazem, está vinculada à programação de computadores por meio do Scratch.

Selby e Woollard (2013), a partir de uma revisão de literatura sobre definições do PC, observam que há autores que não se preocupam em desenvolver uma definição para o termo, porém, ressaltam que há outros que defendem a necessidade desse desenvolvimento, afirmando que, caso não haja uma definição, fica difícil de se desenvolver, por exemplo, uma forma de avaliar o desenvolvimento desse tipo de pensamento no contexto educacional, especialmente se o PC for incorporado ao currículo.

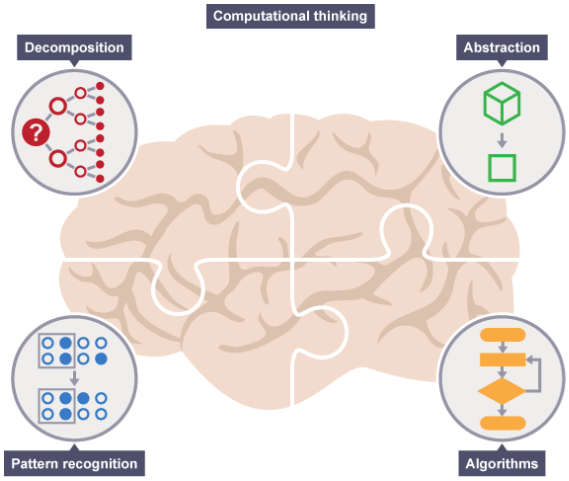
Esses autores observam que parece haver um consenso de que uma definição de pensamento computacional deva incluir a ideia de um processo de pensamento, o conceito de abstração e o conceito de decomposição.

Assim, a partir da revisão, Selby e Woollard (2013, p.5) definem que

> o pensamento computacional é uma abordagem focada na resolução de problemas, incorporando processos de pensamento que utilizam abstração, decomposição, projeto algorítmico, avaliação e generalizações.

A plataforma digital BBC Bitsize (2015) afirma que, antes de resolver um problema utilizando o computador, é preciso compreendê-lo e identificar formas de resolvê-lo. O pensamento computacional também está envolto nesse contexto e auxilia nessa compreensão prévia. De maneira ilustrada, a BBC Bitsize (2015) apresenta as quatro principais técnicas do PC (também chamadas de pilares), conforme apresentamos na Figura 2.

Figura 2 - Pilares do Pensamento Computacional - BBC



Fonte: BBC Bitsize (2015)

De forma descritiva, os pilares do PC são:

> • decomposição - dividir um problema ou sistema complexo em partes menores e mais gerenciáveis;
> • reconhecimento de padrões – procurando semelhanças entre e dentro dos problemas;
> • abstração – concentrando-se apenas nas informações importantes, ignorando detalhes irrelevantes;
> • algoritmos - desenvolver uma solução passo a passo para o problema, ou as regras a seguir para resolver o problema (BBC BITSIZE, 2015).

Destacando que o PC é muito mais do que a programação de computadores, BBC Bitsize (2015) afirma que a programação é o que se diz a um computador o que e como fazer, já o pensamento computacional permite que você descubra exatamente o que dizer ao computador para fazer. Essa afirmação nos mostra que a programação tem um caráter mais pragmático, enquanto o PC enseja uma maior reflexão para além da programação.

De maneira semelhante, Liukas (2015, p.110) afirma que o Pensamento Computacional significa

> pensar nos problemas de uma forma que permita que os computadores os resolvam. O pensamento computacional é algo que as pessoas fazem, não os computadores. Inclui o pensamento lógico e a capacidade de reconhecer padrões, pensar com algoritmos, decompor um problema e abstrair um problema (tradução nossa)

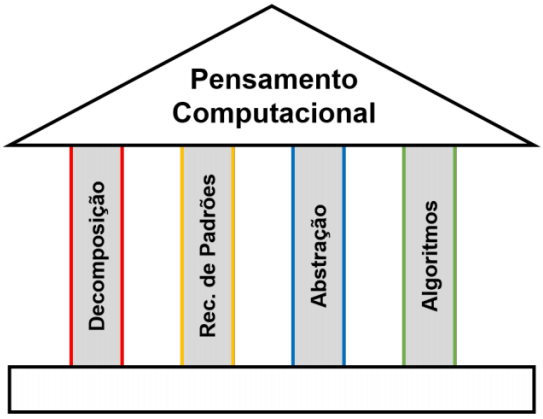
É possível observar na definição para o Pensamento Computacional dessa autora a utilização dos quatro pilares citados por BBC Bitsize (2015). Ela define esses pilares da seguinte forma:

> **Decomposição**: O processo através do qual os problemas são divididos em partes menores. Você pode decompor uma refeição, *cupcakes* ou até níveis de jogo nas partes de que são feitos. Os programadores geralmente dividem seu código em partes menores. Isso torna mais fácil de entender e manter.
> **Reconhecimento de padrões**: Encontrar semelhanças e padrões para resolver problemas complexos de forma mais eficiente. Para encontrar padrões em problemas, procuramos coisas que são iguais (ou muito semelhantes) em cada problema.
> **Abstração**: O processo de separar detalhes que não são necessários para se concentrar nas coisas que são necessárias. Um mapa do metrô é uma abstração do mundo real e complexo. Um calendário é uma abstração do seu tempo. Mesmo as linguagens de programação são abstrações!
> **Algoritmos**: Um algoritmo é um conjunto de etapas específicas que você pode seguir para resolver um problema. [...] Na programação, os algoritmos são usados para criar soluções reutilizáveis para problemas. Mecanismos de busca como Google ou Bing

> usam algoritmos de busca para classificar os resultados (LIUKAS, 2015, p.110-111 – tradução nossa)

Brackmann (2017) também chama esses pilares de dimensões do PC, afirmando que são interdependentes e que trabalham em conjunto com objetivo de resolver problemas. A Figura *3* ilustra a representação desses pilares segundo o autor.

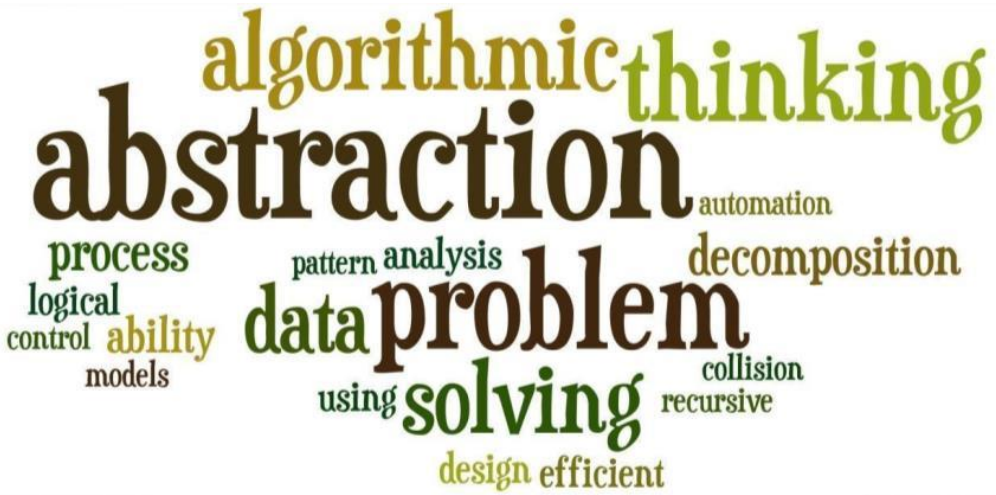
Figura 3 - Pilares do Pensamento Computacional - Brackmann



Fonte: Brackmann (2017, p. 33)

Por meio de uma revisão de literatura, Kalelioglu, Gülbahar e Kukul (2016) comprovaram que realmente ainda não há unanimidade para definir o termo. Os autores buscaram artigos que trouxessem uma definição para pensamento computacional e criaram uma nuvem de palavras com essas variadas definições. Essa nuvem está ilustrada na Figura 4.

Figura 4 - Nuvem de palavras das definições de Pensamento Computacional



Fonte: Kalelioglu, Gülbahar e Kukul (2016, p.584)

É possível observar que abstração, problemas, solução, algoritmo e pensamento são as palavras mais frequentes nas definições dadas nos artigos. Conforme os autores puderam verificar, a maior parte das definições encontradas na literatura ressaltam a resolução, elaboração e entendimento de problemas. Em outras publicações, Wing (2014, 2017) também afirma que o PC está ligado a formular e dar solução a um problema, destacando que essa solução seja dada de forma que um computador também possa "entender".

Em uma tentativa de construir uma definição para o termo, Brackmann (2017), a partir de um levantamento de literatura, o definiu como sendo

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

Para Vicari, Moreira e Menezes (2018), o pensamento computacional é uma metodologia adquirida a partir dos conceitos da Ciência da Computação que deve ser trabalhada de forma interdisciplinar. Os autores ressaltam que o termo está em constante evolução, assim como seus limites e definições, concluindo que era de se esperar que não houvesse uma definição exata para o termo.

Navarro (2021) procura construir uma definição/conceito para o termo Pensamento Computacional no contexto da Educação Matemática. Para isso, faz um levantamento de teses e dissertações que abordem o PC juntamente com a matemática, além da BNCC e do currículo de São Paulo. Em linhas gerais, ela observa que o PC ainda está muito ligado à programação de computadores e propõe que o PC no contexto da Educação Matemática esteja ligado à resolução de problemas, pensamento algébrico e algorítmico, trazendo uma abordagem mais ampla para que o aluno seja protagonista no processo.

Dessa forma, a autora assevera que:

> [...] o pensamento computacional, no âmbito da Matemática escolar, tem a função de auxiliar os alunos a produzirem conhecimentos matemáticos (pensamento algébrico e algorítmico), desenvolverem capacidades de investigação e resolução de problemas, tal qual de leitura de mundo etc. Dessa forma, há de se romper com o paradigma empírico-discursivo, tecnicista e mecanicista, que apregoa o treinamento de técnicas, a memorização de fórmulas e a reprodução de provas e axiomas, mediante, por exemplo, um trabalho pautado na linguagem de programação como produto (NAVARRO, 2021, p.71)

Segundo essa autora, para que o aluno consiga desenvolver o PC a partir de uma situação problema, é necessário que ele perpasse por quatro movimentos, quais sejam: interpretação dos dados, levantamento e sistematização de hipóteses, buscar regularidades e apropriar-se de abstrações. Nesse formato, Navarro (2021) ressalta a importância de não compreender o PC de maneira instrumental e utilitário, mas tê-lo

> como um meio potencial de expandir as capacidades de resolução de problemas, de interpretar a realidade e de ampliar as formas de ação dos alunos diante de seu contexto sociocultural, seja de maneira plugada (TDIC), seja desplugada. Destarte, mais do que pensar "o que é" pensamento computacional [...], é essencial pensarmos "como posso usá-lo no cotidiano" e "como posso aplicá-lo para interpretar e resolver problemas" [...] (NAVARRO, 2021, p.141)

Conforme observamos, o PC é algo amplo e complexo que pode ser trabalhado de diversas formas no contexto escolar. Em sua maioria, conforme já destacamos, ele é desenvolvido por meio da programação de computadores. Tal fato ficou também evidenciado a partir de uma revisão que realizamos dos artigos publicados em um Workshop do Congresso Brasileiro de Informática na Educação (CARVALHO; BRAGA, 2022).

Nessa revisão, pudemos observar que ainda há uma relevante quantidade de atividades que deixam subentendido haver um desenvolvimento espontâneo do Pensamento Computacional a partir da simples manipulação de dispositivos de tecnologia, ou do trabalho com a programação de computadores. Para além disso, pudemos observar ainda que é possível trabalhar o PC a partir da programação de computadores, porém, esse não é o único meio, apesar de ser um dos mais utilizados.

Tal fato ficou comprovado pela revisão de literatura apresentada por Zanetii, Borges e Ricarte (2016). Eles observaram que são cinco as principais formas utilizadas para se trabalhar o Pensamento Computacional, sendo que a programação visual e o trabalho com atividades desplugadas são os dois formatos mais utilizados. A robótica educacional, jogos digitais e linguagem de programação são as outras formas que os autores observaram. Em nossa abordagem, procuramos mesclar entre atividades desplugadas e plugadas, culminando na resolução de um problema por meio da programação visual no *software* Scratch.

A partir da literatura analisada e considerando o contexto de nosso trabalho, entenderemos o PC como a capacidade de resolver um problema, dividindo-o em partes menores, reconhecendo padrões e eliminando partes irrelevantes à elaboração de uma solução em um formato de algoritmo, valendo-se de conceitos computacionais.

Essa compreensão do Pensamento Computacional dará luz a nossas observações da oficina, abrindo oportunidade para nossa discussão em relação a potencialização desse estilo de pensamento no contexto da oficina. Na seção seguinte traremos mais detalhes em relação a avaliação do PC.

2.2 Formato de avaliação do PC

A avaliação do desenvolvimento do PC é alvo de diversas discussões na literatura, não existindo um consenso sobre como avaliá-lo. A maior parte dos trabalhos presentes na literatura trazem um formato de avaliação quantitativo, seja por testes pré/pós intervenção, seja pela análise de códigos de programação e sua quantificação. Há outros autores que optam pelas avaliações em um formato qualitativo, e outros ainda que discutem um formato misto.

Há esforços em busca de um formato de avaliação automático para os códigos desenvolvidos utilizando o *software* Scratch, observando boas práticas de programação. Um exemplo é o Dr Scratch. Ele é uma ferramenta online livre, desenvolvida por Moreno-León e Robles (2015), que avalia o código de projetos desenvolvidos no referido *software*, fornecendo um *feedback* sobre as práticas de programação visualizadas no código. A análise feita pelo *software* retorna uma pontuação em relação a competências relacionadas ao PC (controle de fluxo, representação de dados, abstração, lógica, paralelismo etc), possibilitando que o usuário saiba quais são os pontos que necessita aperfeiçoar em seu código desenvolvido (MORENO-LEÓN; ROBLES, 2015).

Brennam e Resnick (2012) apresentam três tipos de avalição por eles já experimentada para verificar o desenvolvimento do PC de alunos envolvidos com a construção de projetos Scratch.

A primeira abordagem que eles apresentam é a avaliação do portfólio de projetos dos programadores. Acessando a comunidade online Scratch, é possível visualizar os projetos desenvolvidos e coloca-los em uma ferramenta online chamada Scrape. Ela mostra o grau de utilização dos códigos Scratch nos projetos por eles avaliados e, com isso, conseguem visualizar a maturidade da utilização desses códigos pelo programador. É interessante ver como os alunos vão conseguindo avançar na utilização desses códigos, porém, conforme destacado pelos autores, esse tipo de avaliação é limitada, pois está focada apenas no produto, não permitindo que eles tenham qualquer conhecimento em relação ao processo de desenvolvimento.

A segunda abordagem de avaliação comentada por Brennam e Resnick (2012) foi uma entrevista baseada no projeto. Eles abriam alguns projetos do portfólio de um programador e

iam questionando sobre o projeto, sobre a utilização dos códigos etc. Essa entrevista mostrou ainda mais fragilidades da abordagem anterior, pois muitos programadores não sabiam explicar os códigos presentes em seus projetos, o que reforçava o fato de que ter o código em seu programa não significava saber seu funcionamento. Essa abordagem de avaliação foi demasiadamente demorada, além de ser limitada ao projeto discutido que, muitas vezes, era selecionado pelo programador entrevistado por ser mais "legal" e, às vezes, não mostrar momentos de desafios superados.

A terceira e última abordagem para avaliar o Pensamento Computacional foi a aplicação de uma espécie de teste. Nele, os alunos recebiam cenários Scratch já prontos, e os programadores deveriam (1) explicar o que o projeto selecionado fazia, (2) descrever como ele poderia ser estendido, (3) corrigir alguma falha e (4) remixar o projeto adicionando um recurso. Conforme Brennam e Resnick (2012), essa foi uma abordagem interessante por conseguirem apresentar os projetos em um formato crescente de dificuldade, porém, demanda-se muito tempo, especialmente quando os alunos devem corrigir e modificar um projeto, além do mais, por quanto tempo os alunos devem ficar trabalhando nisso? É possivel dar algum suporte aos alunos quando eles apresentam alguma dificuldade?

No geral, os autores concluem que avaliar o desenvolvimento do PC não é uma tarefa simples, e que nenhuma das três abordagens se mostrou totalmente satisfatória, mas talvez uma mistura entre elas seja mais adequada. Vale reforçar que essas avaliações apresentadas por Brennam e Resnick (2012) estavam relacionadas estritamente à programação feita no Scratch.

A revisão de literatura desenvolvida por Araújo, Andrade e Guerrero (2016) com objetivo de buscar conhecer um pouco mais em relação à avaliação do PC, mostra que o método avaliativo mais utilizado é o teste, sendo que nele estão incluídos questionários pré e pós intervenção, bem como provas do PISA e ENEM. Além de testes, há também a análise dos códigos de programação e projetos como formas de avaliar, observando as estruturas da programação e processos de desenvolvimentos. Em menor quantidade os autores encontraram avaliação observacional, que incluíram discussão das atividades e análise dos artefatos (ARAÚJO; ANDRADE; GUERRERO, 2016). Os autores destacam que há avaliações que utilizam granularidade alta na avaliação (consideram apenas a nota final de um teste), ressaltando que tal fato dificulta o avaliador identificar qual habilidade o estudante apresenta déficit. Os autores apontam que uma granularidade fina (avaliar separadamente as habilidades) pode minimizar essa questão, apesar de tornar a avaliação mais custosa.

Barcelos, Bortoletto e Andrioli (2016) ofereceram, em um projeto de extensão, uma oficina online para o desenvolvimento do PC a egressos ou graduandos, sendo a maior parte

dos participantes da área de matemática. Como forma de avaliar o desenvolvimento do PC, os autores utilizaram rubricas que determinaram se o aluno alcançou o nível básico, intermediário ou avançado em cada conceito do PC no projeto final do curso.

Barcelos *et al.* (2017) apresentam outro formato de avaliação automática utilizando mapas auto-organizáveis a partir de dados gerados em oficinas para desenvolvimento de jogos digitais. Conforme os autores, essa análise busca identificar estratégias utilizadas para a resolução dos problemas propostos.

Raabe *et al.* (2017), por meio de um jogo, exploraram questões baseadas em testes de QI e Questões do Programa de Enriquecimento Instrumental de Feuerstein para avaliar o PC. Em cada uma das fases do jogo, elementos do PC foram explorados com base nos termos da Associação de Professores de Ciência da Computação (CSTA). Os autores ressaltam que o objetivo não é a construção de uma nota final para o aluno, mas sim identificar o nível de desenvolvimento do aluno para cada um dos elementos do PC.

Em sua pesquisa, Brackmann (2017) avaliou habilidades do PC em estudantes espanhóis e brasileiros a partir de um instrumento chamado de Teste de Pensamento Computacional. Esse teste está validado para quatro habilidades do pensamento computacional: decomposição, reconhecimento de padrões, abstração e pensamento algoritmo por meio de 28 questões. Com base nos acertos dos alunos dessas questões, por meio de métodos estatísticos, é observado o conhecimento dos alunos.

Menezes (2022) traz uma rica discussão qualitativa em torno da construção/desenvolvimento do Pensamento Computacional a partir da cidade, com explorações de diversos ambientes culturais. A autora procurou desenvolver com os alunos atividades a partir de visitações a diferentes locais, explorando não só o Pensamento Computacional, mas também a questão cultural e histórica, além de explorar o idioma inglês. Conforme a autora, o PC se dá nos diversos processos de interação entre humanos e não humanos, em um movimento de experiências com e na cidade.

A partir da literatura analisada em relação à avaliação do PC, pudemos observar que não há um consenso em relação a tal e, portanto, em nossa abordagem, utilizaremos uma observação processual da oficina, voltando nossa atenção para a utilização dos conceitos computacionais apresentados por Brennam e Resnick (2012), bem como para o trabalho com os quatro pilares do PC (LIUKAS, 2015; BBC BITSIZE, 2015; BRACKMANN, 2017). Traremos mais detalhes sobre o formato de nossa avaliação no capítulo referente à metodologia da pesquisa.

A seguir apresentaremos um pouco da teoria construcionista de Papert.

2.3 Construcionismo de Papert

O termo construcionismo foi cunhado pelo educador estadunidense Seymour Papert. Entusiasta na inserção dos computadores no contexto escolar, Papert desenvolveu a linguagem de programação Logo na década de 1960, objetivando levar à escola um meio para que os alunos pudessem ser os próprios construtores de seu conhecimento mediados pelo computador.

Essa linguagem de programação consiste em "ensinar" uma tartaruga virtual a desenvolver determinadas tarefas, por exemplo, construir um quadrado. Para tanto, é necessário "dizer" a ela quantos passos deve dar para frente, quantos graus girar e assim por diante. À medida que a tartaruga vai se deslocando a partir dos comandos recebidos, ela vai riscando por onde passa e então o desenho vai sendo gerado. O autor acredita que ao ensinar a tartaruga a fazer essas tarefas, a criança desenvolve habilidades que podem auxiliar em seu crescimento intelectual.

Para Papert (1994), ao invés de afirmarmos que estamos na Era da Informática, deveríamos afirmar que estamos na Era da Aprendizagem, visto que muitas possibilidades se abrem com o advento do computador e da internet. Ele ressalta que

> hoje em dia, nos países industrializados, a maioria das pessoas tem empregos que não existiam quando elas nasceram. A habilidade mais importante na determinação do padrão de vida de uma pessoa já se tornou a capacidade de aprender novas habilidades, de assimilar novos conceitos, de avaliar novas situações, de lidar com o inesperado (PAPERT, 1994, p.5).

Nesse formato, desenvolver a habilidade de pensar melhor e solucionar problemas são prementes em um mundo no qual as mudanças acontecem de forma rápida. Para tanto, Papert (1994) afirma que é muito importante estar conectado com aquilo que se desenvolve, trazendo seu próprio corpo à baila. Na utilização da linguagem Logo, por exemplo, é importante que a criança se imagine sendo a tartaruga, nesse formato ela conseguirá melhor compreender a questão dos passos e giros a serem dados para conseguir construir as formas geométricas.

Toda essa ideia de construção de formas geométricas no contexto da programação de computadores por meio da linguagem Logo foi pautada na filosofia construcionista que, conforme Papert (1994) deixa claro, é oposta à instrucionista.

O instrucionismo, segundo o autor, significa algo muito diferente de pedagogia, da arte de ensinar. Em suas palavras, essa filosofia "deve ser lida num nível mais ideológico ou programático, como expressando a crença de que a via para uma melhor aprendizagem deve ser

o aperfeiçoamento da instrução – se a Escola é menos que perfeita, então sabemos o que fazer: ensinar melhor" (PAPERT, 1994, p.124).

Já o construcionismo é uma filosofia que, apesar de não colocar em dúvida o valor da instrução, nega a verdade sobre melhorá-la afim de produzir melhor aprendizado. Assim, "a meta é ensinar de forma a produzir a maior aprendizagem a partir do mínimo de ensino" (PAPERT, 1994, p.125). Nesse formato, Papert reforça que não basta apenas reduzir o ensino e deixar todo o resto de lado, mas é preciso ensinar a pescar ao invés de dar o peixe.

Em relação ao construcionismo, Maltempi (2012) afirma que ele

> é tanto uma teoria de aprendizado quanto uma estratégia para a educação que compartilha a ideia de que o desenvolvimento cognitivo é um processo ativo de construção e reconstrução das estruturas mentais, no qual o conhecimento não pode ser simplesmente transmitido do professor para o aluno (MALTEMPI, 2012, p.288)

Sob essa perspectiva, destaca-se que o processo de aprendizagem em um contexto construcionista é ativo, de movimento, no qual o aluno não fica parado apenas ouvindo o professor, mas busca por si, produz, constrói. Valente (2005b) ressalta que a simples inserção do computador no contexto escolar não significa um trabalho construcionista. Conforme ele relembra as ideias de Papert (1994), muitas vezes o computador é utilizado como um simples meio de digitalizar os formatos tradicionais de ensino, ou seja, ao invés de o professor dar as instruções aos alunos, quem o faz é o computador. Nesse formato, o paradigma instrucionista segue o mesmo, porém, as instruções são passadas pelo computador e não mais pelo professor.

Valente (2005b) destaca que o construcionismo que Papert (1994) propõe é muito diferente disso. A ideia é que o aluno possa criar coisas, construir por meio do computador. É nesse sentido que Papert (1994) desenvolveu a linguagem Logo para que os alunos pudessem fazer suas construções, resolver problemas utilizando o computador.

Nesse contexto é importante destacar que um ambiente construcionista não depende exclusivamente da programação de computadores. Maltempi (2012) ressalta que, um processador de textos, por exemplo, pode ser utilizado de forma construcionista, permitindo uma aprendizagem ativa. Não só o processador de textos, mas "diversas ferramentas computacionais existentes podem ser consideradas construcionistas se forem empregadas de maneira adequada" (MALTEMPI, 2012, p.289).

Conforme Papert (1994) deixa claro, sua filosofia construcionista foi baseada no construtivismo de Piaget, porém, para ele, é importante o desenvolvimento de um artefato. Em suas palavras,

> o Construcionismo, minha reconstrução pessoal do Construtivismo, apresenta como principal característica o fato de que examina mais de perto do que outros – ismos educacionais a ideia da construção mental. Ele atribui especial importância ao papel das construções no mundo como um apoio para o que ocorreu na cabeça, tornando-se desse modo, menos uma doutrina puramente mentalista (PAPERT, 1994, p.127-128)

Ao construir um artefato "no mundo", ou seja, alguma construção que pode ser apresentada, como uma torta, uma casa feita de Lego, um programa de computador, Papert (1994) acredita que há um maior crescimento intelectual, visto que, ao construir esse artefato, a criança está transformando sua construção mental para um formato que pode ser discutido. Para Papert, essa construção é um dos seus princípios matéticos[10].

A partir de um longo tempo de estudo em relação aos ambientes de programação Logo, cinco importantes dimensões para esse tipo de ambiente são destacadas, quais sejam:

> 1. Dimensão pragmática: refere-se à sensação que o aprendiz tem de estar aprendendo algo que pode ser utilizado de imediato, e não em um futuro distante. [...]
> 2. Dimensão sintônica: ao contrário do aprendizado dissociado, normalmente praticado em salas de aula tradicionais, a construção de projetos contextualizados e em sintonia com o que o aprendiz considera importante fortalece a relação aprendiz-projeto, aumentando as chances de que o conceito trabalhado seja realmente aprendido. [...]
> 3. Dimensão sintática: diz respeito à possibilidade de o aprendiz facilmente acessar os elementos básicos que compõem o ambiente de aprendizagem e progredir na manipulação destes elementos de acordo com a sua necessidade e desenvolvimento cognitivo. [...]
> 4. Dimensão semântica: refere-se à importância de o aprendiz manipular elementos que carregam significados que fazem sentido para ele, em vez de formalismos e símbolos. [...]
> 5. Dimensão social: aborda a integração da atividade com as relações pessoais e com a cultura do ambiente no qual ele se encontra. O ideal é criar ambientes de aprendizagem que utilizem materiais valorizados culturalmente. Nesse sentido, a programação de computadores e o domínio da tecnologia em geral representam bons materiais a serem aproveitados, uma vez que são bem valorizados na sociedade atual. [...]
> (MALTEMPI, 2012, p. 290-291)

Maltempi (2012) reforça que essas dimensões devem ser buscadas quando se propõe o trabalho com o construcionismo, ressaltando que essa filosofia vai além de aspectos cognitivos, mas procura tangenciar também questões relacionadas à tecnologia, personalidade, cultura e motivação.

A ideia construcionista de Papert serviu de base para nossa oficina, especialmente na parte final, na qual os alunos deveriam resolver os problemas propostos por meio do

[10] Matética é uma palavra cunhada por Papert para indicar a capacidade de aprender. Essa palavra procura trazer uma oposição à pedagogia, que segundo ele, seria a capacidade de ensinar.

desenvolvimento de um programa no Scratch. Os detalhes dessa oficina serão apresentados no capítulo seguinte, referente aos procedimentos metodológicos, na seção sobre a produção dos dados.

2.4 Teses correlatas

Com objetivo de visualizar outras pesquisas de doutorado que trabalharam com o Pensamento Computacional no contexto da Educação Básica, buscando relacionar nosso trabalho com essas outras pesquisas, fizemos uma busca por teses no Catálogo de Teses e Dissertações da Capes[11], utilizando a chave a chave de busca "Pensamento Computacional" AND "Educação Básica", o que nos retornou 8 trabalhos. É importante destacar que essa busca aconteceu no mês de maio de 2023.

Os trabalhos retornados na busca estão apresentados no Quadro *1*.

Quadro 1 - Teses que relacionam Pensamento Computacional e Educação Básica

Nº	**Título**	**Autor (a)**	**Foi incluído em nossa análise?**
1	Um estudo sobre os estilos de aprendizagem, inteligência lógica e Pensamento Computacional de alunos do Ensino Fundamental e Médio de uma escola colombiana	KATERINE EDITH TOBIO GUTIERREZ	Sim
2	Desenvolvimento do Pensamento Computacional através de Atividades Desplugadas na Educação Básica	CHRISTIAN PUHLMANN BRACKMANN	Sim
3	O desenvolvimento do Pensamento Computacional na Educação Matemática segundo contribuições da teoria Histórico-Cultural	ELOISA ROSOTTI NAVARRO	Sim
4	O Pensamento Computacional na resolução de problemas na matemática: Uma proposta de aplicação metodológica integrada na Educação Básica	SAMANTA BUENO DE CAMARGO CAMPANA	Sim

[11] https://catalogodeteses.capes.gov.br/catalogo-teses/#!/

5	Pensamento computacional em livros didáticos do ensino médio: sobre atividades e possibilidades	ENZO BERTAZINI	Sim
6	PCOMP-MODEL: Desenvolvendo o Pensamento Computacional na Educação Básica para auxiliar na aprendizagem de algoritmos e programação no Ensino Superior	MARIA CLAUDETE SCHORR	Sim
7	Práticas pedagógicas: tendências à luz dos estilos de aprendizagem	ELISANGELA APARECIDA BULLA IKESHOJI	Não
8	Design de Games para a aprendizagem: análise das contribuições dos estudantes da Educação Profissional	DALVA CELIA HENRIQUES ROCHA GUAZZELLI	Não

Fonte: o autor

O texto 7, intitulado "Práticas pedagógicas: tendências à luz dos estilos de aprendizagem", não foi considerado em nossa análise, visto que se trata de uma investigação sobre os estilos de aprendizagem de professores e alunos, não havendo uma relação direta com o Pensamento Computacional no contexto da Educação Básica.

Já o texto 8, intitulado "Design de Games para a aprendizagem: análise das contribuições dos estudantes da Educação Profissional", também não foi considerado em nossa análise por apresentar uma pesquisa que analisa as contribuições do design de games nas aulas das disciplinas de programação web e design digital, componentes de um curso técnico em desenvolvimento de sistemas, não havendo uma relação direta com o Pensamento Computacional no contexto da Educação Básica.

A seguir passamos a tecer alguns comentários sobre os demais textos que fizeram parte do nosso rol de análise.

2.4.1 Texto 1 - Um estudo sobre os estilos de aprendizagem, inteligência lógica e Pensamento Computacional de alunos do Ensino Fundamental e Médio de uma escola colombiana

O texto 1, intitulado "Um estudo sobre os estilos de aprendizagem, inteligência lógica e Pensamento Computacional de alunos do Ensino Fundamental e Médio de uma escola colombiana", procura estudar a relação de três variáveis: Estilos de Aprendizagem – EA –,

Inteligência Lógica Matemática – ILM – e Pensamento Computacional – PC. Essas variáveis são consideradas importantes no processo de ensino e aprendizagem no campo da Educação Matemática e Ciência da Computação.

Para fazer seu estudo, Gutierrez (2021) balizou sua pesquisa pela pergunta: Como se relacionam os Estilos de Aprendizagem, Inteligência Lógica Matemática e Pensamento Computacional entre si e com as variáveis independentes (ano escolar, gênero e desempenho acadêmico de Matemática e Informática) dos estudantes do Ensino Fundamental e Ensino Médio de uma escola pública do Município de Corozal, Sucre, Colômbia?

Para alcançar o objetivo de analisar as relações entre os Estilos de Aprendizagem (EA), Inteligência Lógica Matemática (ILM) e o Pensamento Computacional (PC), a autora se valeu de uma pesquisa quantitativa, aplicando questionários a 360 estudantes colombianos do Ensino Fundamental II e Médio e fazendo a análise estatística dos resultados.

O primeiro questionário, composto por 80 itens, procurou avaliar o estilo de aprendizagem dos alunos (Ativo, Reflexivo, Teórico, Pragmático). O segundo questionário, composto por 50 itens, foi o Teste de Inteligência Lógica e, por fim, o terceiro questionário, com 20 questões, foi o teste de PC.

Por meio de métodos estatísticos, a autora analisou os dados e pôde observar, em linhas gerais, que há uma correlação entre o PC e a Inteligência Lógica Matemática. Gutierrez (2021), afirma que uma potencialização do PC perpassa também pela potencialização da inteligência lógico matemática.

Destacamos que nossa pesquisa se distancia da apresentada pela autora, visto que nossa proposta está ancorada na potencialização do PC por meio da oficina híbrida a partir de uma análise qualitativa das produções dos alunos; a pesquisa aqui analisada, por meio de uma análise quantitativa, observa a correlação entre os estilos de aprendizagem, a inteligência lógica matemática e o PC.

Em relação à conclusão que a autora chegou com sua pesquisa, apontamos que, ao procurarmos potencializar o PC nos alunos a partir da oficina híbrida, estamos também contribuindo para a potencialização da lógica matemática, e vice versa. Dessa forma, ao trabalharmos a questão da resolução de problemas por meio do PC, estamos potencializando nos alunos também a matemática e a possibilidade de resolução de problemas das mais diversas áreas.

2.4.2 Texto 2 - Desenvolvimento do Pensamento Computacional através de Atividades Desplugadas na Educação Básica

O texto 2, intitulado “Desenvolvimento do Pensamento Computacional através de Atividades Desplugadas na Educação Básica”, apresenta uma pesquisa em relação ao desenvolvimento do PC de forma desplugada.

Brackmann (2017) procurou desenvolver seu trabalho com alunos brasileiros e espanhóis dos anos iniciais do Ensino Fundamental II (crianças de 10 a 12 anos). Ele trabalhou uma série de atividades no papel com os alunos, usando cartas, jogos de tabuleiro etc.

O autor justifica seu foco no trabalho exclusivamente desplugado (trabalhar PC sem a utilização de recursos digitais) na questão do alcance de sua pesquisa, visto que, dessa forma, é possível levar esse trabalho a qualquer escola, independente de haver recursos digitais, internet e até mesmo energia elétrica.

Após as aplicações das atividades, por meio de um instrumento chamado de Teste de Pensamento Computacional que contém 28 questões, o autor avaliou o desenvolvimento dos alunos nas quatro habilidades (pilares) do PC: decomposição, reconhecimento de padrões, abstração e pensamento algoritmo.

Com base no resultado dos alunos nesse teste, por meio de métodos estatísticos, o autor pôde verificar uma melhoria significativa no desempenho dos estudantes, tanto brasileiros quanto espanhóis.

Ao analisarmos a pesquisa aqui apresentada, observamos que, assim como o nosso trabalho, o autor trabalhou o PC com alunos da Educação Básica, porém, utilizamos um formato híbrido para nossa oficina, entendendo que, ao trabalhar as atividades também no formato plugado, temos novas possibilidades de exploração que só são possíveis dessa maneira, como por exemplo o caráter exploratório, o *feedback* imediato proporcionado pelas tecnologias digitais.

Outro ponto que difere de nossa pesquisa está na análise de seus dados. Conforme o autor apresenta, ele utiliza um método quantitativo de análise a partir da aplicação de um teste, enquanto que em nossa pesquisa nos valemos de uma análise qualitativa, valorizando mais a descrição e a manifestação dos sujeitos ao longo da aplicação.

2.4.3 Texto 3 - O desenvolvimento do Pensamento Computacional na Educação Matemática segundo contribuições da teoria Histórico-Cultural

O terceiro texto, intitulado “O desenvolvimento do Pensamento Computacional na Educação Matemática segundo contribuições da teoria Histórico-Cultural”, apresenta a pesquisa de Navarro (2021). Ela procura construir uma definição/conceito para o termo Pensamento Computacional no âmbito da Educação Matemática. Para tanto, faz um levantamento de teses e dissertações no período de 2009 a 2019 que abordem o PC juntamente com a matemática, além de tomar como base a BNCC e do currículo do estado de São Paulo.

Em linhas gerais, tomando como pressuposto a Teoria Histórico-Cultural, ela observa que o PC ainda está muito ligado à programação de computadores. A partir dessa observação, ela aponta que o PC no contexto da Educação Matemática está ligado à resolução de problemas, pensamento algébrico e algorítmico, trazendo uma abordagem mais ampla, possibilitando assim que o aluno seja protagonista no seu processo de aprendizagem.

Nesse sentido, Navarro (2021) reforça que o PC no contexto da educação matemática é constituído pela ligação inseparável entre os nexos conceituais da resolução de problemas e dos pensamentos algébrico e algorítmico. Desse modo, ela aponta que o PC nesse contexto auxilia os alunos não só na construção de conhecimentos matemáticos, mas também no desenvolvimento de capacidades de investigação e resolução de problemas.

A autora conclui sustentando a necessidade de se compreender o PC não apenas de modo instrumental, unicamente trabalhado a partir das tecnologias digitais, mas o defende como um meio para expandir a capacidade de resolução de problemas, seja de maneira plugada ou desplugada.

Nesse viés, observamos nossa pesquisa corroborando com a afirmação de Navarro (2021), visto que, ao propormos uma oficina no formato híbrido, entendemos que tanto atividades plugadas e desplugadas se retroalimentaram e complementaram, possibilitando aos alunos uma visão e experimentação mais ampla do PC.

2.4.4 Texto 4 - O Pensamento Computacional na resolução de problemas na matemática: Uma proposta de aplicação metodológica integrada na Educação Básica

O texto 4 intitulado “O Pensamento Computacional na resolução de problemas na matemática: Uma proposta de aplicação metodológica integrada na Educação Básica” visa apresentar o PC como um instrumento possível para o desenvolvimento das habilidades necessárias ao aluno na aplicação da resolução de problemas que envolvem matemática. Assim, o objetivo de sua pesquisa é investigar como os elementos do PC são operados pelos alunos na resolução de problemas matemáticos.

Campana (2022) ofereceu um minicurso de 9 aulas com objetivo de explorar o PC, resolução de problemas e conhecimento sobre a linguagem de programação *processing*. Conforme ela relata, a parte inicial do minicurso (3 aulas) aconteceu de forma desplugada. Nessa parte, os alunos trabalharam com a torre de Hanói. Já as outras 6 aulas do minicurso aconteceram de forma plugada, utilizando o *processing*, resolvendo problemas de matemática pela programação de computadores.

O minicurso foi destinado a alunos do nono ano do EF II que tiveram interesse em participar, sendo ele ofertado no período contrário ao horário regular das aulas. Conforme relatado pela autora, apenas 8 alunos participaram.

A análise dos dados feita por Campana (2022) aconteceu de forma qualitativa, sendo as atividades que os alunos desenvolveram bem como alguns questionários por ela aplicados que compuseram o rol de dados para a análise. Ela conclui que houve a utilização de habilidades do PC na resolução de problemas, destacando que o PC se mostrou um importante aliado para a melhora na resolução de problemas de matemática.

Nossa pesquisa se aproxima da apresentada quando a autora desenvolve atividade plugada e desplugada e destaca a positividade da associação de ambas as formas de abordagem, porém, o foco da presente pesquisa está na resolução de problemas matemáticos, procurando articular o PC a essa disciplina, o que se distancia de nossa proposta.

Em nossa pesquisa procuramos propor uma oficina no formato híbrido, que buscou articular atividades plugadas e desplugadas em um formato incremental de dificuldade, potencializando o PC. Vale destacar que nosso trabalho envolveu toda a turma, e não um grupo restrito. Dessa forma, nossa proposta apresenta um potencial formato de trabalho que pode ser desenvolvido na rotina escolar.

É importante destacar ainda que, a linguagem de programação *processing* não possui uma interface de programação visual, ou seja, os códigos de programação precisam ser digitados. Conforme relatado pela autora, nessa etapa de seu trabalho foi necessário ter uma pessoa auxiliando com os códigos de programação. Já a linguagem Scratch, utilizada em nossa pesquisa, por ser visual, lúdica e intuitiva, permite um trabalho mais autônomo dos alunos.

2.4.5 Texto 5 - Pensamento computacional em livros didáticos do ensino médio: sobre atividades e possibilidades

No texto de número 5, intitulado “Pensamento computacional em livros didáticos do ensino médio: sobre atividades e possibilidades”, o autor procura analisar questões que possam contribuir para o desenvolvimento do PC nos alunos.

O autor inicia seu texto relatando a dificuldade dos alunos na disciplina de lógica de programação da licenciatura em matemática em que trabalha e, assim, levanta a questão: que possibilidades para desenvolver o Pensamento Computacional há em atividades propostas em livros aprovados pelo PNLD para o Ensino Médio?

Bertazini (2022), se valendo de uma pesquisa bibliográfica qualitativa, analisou duas coleções de livros para o Ensino Médio, buscando por atividades que oferecessem possibilidades para o desenvolvimento do PC. Em sua análise, identificou atividades que foram abordadas com o uso dos *softwares* GeoGebra e *Scratch.*

O autor comenta rapidamente sobre a existência do trabalho desplugado para o desenvolvimento do PC, porém, ressalta que seu foco está no pensamento computacional plugado e seu desenvolvimento nas atividades presentes nos livros didáticos.

Ele utiliza os pilares do PC – que chama de conceitos (decomposição, o reconhecimento de padrões, a abstração e o pensamento algorítmico) para analisar as questões dos livros didáticos, procurando observar a exploração desses pilares nessas questões.

As atividades escolhidas para análise foram aquelas que mencionavam a utilização dos softwares GeoGebra (geometria dinâmica) e *Scratch* (construção de histórias e animações por meio da programação visual). A escolha por atividades que sugeriam a utilização desses softwares se deu, segundo Bertazini (2022), por possuírem características mais construtivas, que estão mais aderentes à possibilidade de desenvolvimento do PC.

O autor, em sua análise, observa que as questões que sugerem a utilização do GeoGebra, em sua maioria, solicitam ao aluno que reproduza uma sequência de passos (algoritmo) para fazer determinada construção no software, deixando a atividade de modo mais direcionada. Já as atividades que sugerem a utilização do Scratch, trazem um programa exemplo e solicitam alguma resolução dos alunos, deixando margem para uma construção mais livre dos alunos, possibilitando a resolução de formas diferentes.

Nesse viés, Bertazini (2022, p.171) conclui que “o GeoGebra depende dos conhecimentos matemáticos do aluno, mesmo que os conceitos básicos de geometria já estejam associados às ferramentas disponíveis em seus menus”. Em relação ao software S*cratch,* ele conclui que ele “disponibiliza blocos funcionais pré-definidos e sua utilização depende do

raciocínio lógico e da capacidade de abstração do aluno". O autor finaliza suas considerações ressaltando que ambos os softwares analisados possuem possibilidades de auxiliarem no desenvolvimento do PC, porém, observa que, do modo como foi apresentado nos livros didáticos, o *Scratch* se mostrou mais potencial devido ao caráter mais aberto das atividades, que possibilita construções mais livres dos alunos.

As considerações de Bertazini (2022) em relação a suas observações reforçam o caráter potencializador do PC de nossa proposta. Ao trabalharmos a oficina híbrida utilizando o Scratch como recurso digital para a resolução dos problemas, permitimos aos alunos desenvolverem sua criatividade e construção de animações que solucionassem o que propusemos, abrindo possibilidade para a potencialização do PC.

Observamos uma semelhança de nosso trabalho com o de Bertazini (2022) no que diz respeito a análise das questões. Conforme apresentado pelo autor, sua observação se direcionou a utilização dos pilares do PC nas atividades propostas. Um olhar voltado para os pilares do PC também se deu em nossa análise, porém, procuramos complementar ainda com a exploração dos conceitos computacionais. Os detalhes dessa análise serão apresentados no capítulo 3, referente a metodologia de nossa pesquisa.

2.4.6 Texto 6 - PCOMP-MODEL: Desenvolvendo o Pensamento Computacional na Educação Básica para auxiliar na aprendizagem de algoritmos e programação no Ensino Superior

O sexto e último texto analisado é intitulado "PCOMP-MODEL: Desenvolvendo o Pensamento Computacional na Educação Básica para auxiliar na aprendizagem de algoritmos e programação no Ensino Superior". Nele, Schorr (2020) inicia apresentando uma vasta revisão de literatura, destacando a importância do desenvolvimento do PC desde a Educação Básica. Sua tese consiste na apresentação de um modelo (sistema) que apoia o processo de ensino e aprendizagem de habilidades relacionadas ao PC.

Sua pesquisa foi norteada pela questão: Como um modelo de apoio ao ensino e aprendizagem pode instigar a adoção de comportamentos adequados nos estudantes do ensino médio para o desenvolvimento das competências para aprendizagem de Algoritmos e Programação?

O modelo de ensino e aprendizagem desenvolvido pela autora foi denominado de PComp-Model. Esse modelo, conforme apresentado, possui duas seções. A primeira é destinada ao professor, servindo como um guia para a elaboração de exercícios que explorem o PC. A segunda é destinada ao aluno, procurando orientá-lo na resolução dos problemas.

Conforme ela destaca, sua proposta vai além da utilização de ferramentas para o desenvolvimento do PC, procurando apoiar o desenvolvimento de competências para o PC, como a resolução de problemas.

Para elaborar seu modelo, a autora aplicou atividades a alunos da graduação na área de informática para verificar suas principais dificuldades na resolução dos problemas. O resultado dessa aplicação auxiliou na elaboração do PComp-Model. Esse modelo foi desenvolvido em um sistema Web, no qual professor e aluno têm acesso por meio de senha para elaborar e resolver as questões, respectivamente.

Procurando validar seu modelo, a autora o aplicou a 4 professores (de diferentes disciplinas) e 21 alunos do terceiro ano do Ensino Médio, sendo esses alunos divididos em dois grupos, um deles passando pelo modelo PComp-Model e outro não. Depois dessa etapa, todos participaram de uma oficina de programação e seus resultados foram comparados.

Schorr (2020) observou que o grupo que havia passado pelo modelo PComp-Model apresentou melhores soluções, sendo que tiveram mais facilidade para interpretar os problemas e identificar os elementos necessários à solução.

Analisando o trabalho da autora, observamos uma aproximação com nossa proposta no sentido de trabalhar o PC relacionado com a resolução de problemas, porém, nos distanciamos dela quando sua proposta visa a elaboração de um modelo de construção/resolução desses problemas, visto que nosso trabalho procurou apresentar uma oficina híbrida para a potencialização do PC. Tanto a proposta de nossa tese quanto a da autora analisada procura apresentar um meio para desenvolvimento do PC, porém, a proposta da autora está em auxiliar os professores na elaboração de questões em um formato que os alunos utilizem o PC na solução e, em nossa proposta, apresentamos uma oficina no formato híbrido que procura potencializar o PC nos alunos.

Ao finalizarmos essa análise das 6 teses publicadas no banco de teses e dissertações da Capes que, de alguma maneira, relacionaram o Pensamento Computacional com a Educação Básica, pudemos concluir que nosso trabalho traz uma relevante contribuição para o trabalho com o Pensamento Computacional na Educação Básica.

Nossa proposta apresenta um formato inovador de oficina, que busca mesclar atividades desplugadas e plugadas, em um movimento incremental de dificuldade, que vai permitindo aos alunos observarem o PC em um formato lúdico, relacionando-o ao cotidiano e à resolução de problemas em um formato algorítmico.

Além do formato híbrido da oficina, apresentamos ainda um estilo de análise que contribui para a reflexão da comunidade científica, visto que nossa análise busca observar a

articulação dos pilares do PC com os conceitos computacionais. Os detalhes dessa análise, bem como os demais procedimentos metodológicos estão detalhes no capítulo que segue.

3 PROCEDIMENTOS METODOLÓGICOS

Nesse capítulo apresentamos de maneira detalhada nossa postura metodológica para realizar o presente estudo. Apontaremos, em linhas gerais, qual o tipo de pesquisa adotado, procurando trazer algumas classificações, porém, ressaltamos que muito mais do que classificar nossa pesquisa, queremos apresentar de forma detalhada quais os passos tomados ao longo do processo, que, conforme André (2013), é o que traz mais clareza sobre o objeto de estudo, aumentando o rigor da pesquisa.

3.1 Sobre a postura metodológica

O fenômeno a ser estudado em uma pesquisa é o que determina se o paradigma qualitativo ou o quantitativo será mais indicado. Para um objeto passível de mensuração, trabalhar com o paradigma quantitativo pode ser mais adequado, enquanto, para um objeto mais descritivo, utilizar o paradigma qualitativo se enquadra melhor. Garnica (1997, p. 111) destaca que, uma pesquisa do tipo qualitativa consiste em

> uma trajetória circular em torno do que se deseja compreender, não se preocupando única e/ou aprioristicamente com princípios, leis e generalizações, mas voltando o olhar à qualidade, aos elementos que sejam significativos para o observador-investigador.

Em contrapartida, as pesquisas de cunho quantitativo preconizam pelo rigor objetivo e a sistematização. Bicudo (2013) ressalta que o quantitativo está relacionado ao que pode ser mensurável. Conforme a autora,

> ele carrega consigo as noções próprias ao paradigma positivista, que destaca pontos importantes para a produção da ciência, a razão, a objetividade, o método, a definição de conceitos, a construção de instrumentos para garantir a objetividade da pesquisa (BICUDO, 2013, p. 115).

Padrões, regras para realizar testes precisos, normalmente de fundo matemático, são preconizadas pela pesquisa quantitativa (GARNICA, 1997), enquanto descrições e observações prevalecem na pesquisa qualitativa.

A partir dessas considerações, observamos que nossa pesquisa está ancorada na base qualitativa, visto que nosso fenômeno de estudo está no âmbito da construção de conhecimento dos alunos em uma oficina de Pensamento Computacional a partir de suas discussões e argumentações. Como nossa intenção não está na aplicação de avaliações para verificar acertos e

erros, mensurar o conhecimento dos alunos por meio de notas, a pesquisa quantitativa não daria conta daquilo que almejamos, visto que ela seria insuficiente para descrever as nuances do processo de ensino e aprendizagem envolvidas nessa proposta, pois os dados gerados na pesquisa não são essencialmente mensuráveis, mas sim passíveis de discussão a partir de observações do pesquisador ao longo do processo.

Dessa forma, estamos alinhados ao que afirmam Lüdke e André (2018), de que no contexto do paradigma qualitativo de pesquisa, a análise descritiva dos dados coletados é priorizada; o processo tem maior relevância do que o produto; a indução é mais valorizada para a análise dos dados.

Na condução de nossa pesquisa, optamos por trabalhar à luz do estudo de caso. Esse é um formato de pesquisa que busca desenvolver um estudo profundo e detalhado sobre determinado caso (GIL, 2008). Conforme destacado por Ponte (2006, p. 2), esse tipo de pesquisa qualitativa "visa conhecer uma entidade bem definida como uma pessoa, uma instituição, um curso, uma disciplina, um sistema educativo, uma política ou qualquer outra unidade social". Assim, o autor reforça que o objetivo do estudo de caso é compreender de forma aprofundada essa entidade.

Ponte (2006) ainda ressalta, com base em Yin (2006), que os estudos de caso podem ter diferentes propósitos, quais sejam: exploratórios, servindo para obter informação preliminar do objeto; descritivos, tendo como propósito dizer “como é” o caso em análise; e analíticos, que procura problematizar o objeto ou desenvolver nova teoria, confrontando-a, ou não, com teoria já existente.

A partir de uma discussão entre as teorias sobre estudo de caso de Robert Yin e Robert Stake, Alves-Mazzotti (2006) aponta que há muitas pesquisas rotuladas como estudo de caso que não poderiam ser assim caracterizadas. Conforme a autora, há pesquisas isoladas que simplesmente olham para determinado caso sem ter qualquer articulação com a teoria já existente, sem se preocupar em contextualizar a pesquisa no âmbito da comunidade acadêmica, tornando assim aquele estudo de caso isolado e que pouco contribui para o avanço da ciência. Conforme destacado, é importante haver essa "conversa" com a comunidade científica, oportunizando uma avaliação crítica dos pares e reforçando assim a preocupação com a construção coletiva do conhecimento.

Lüdke e André (2018, p. 18-19) apresentam importantes características de um estudo de caso qualitativo, quais sejam:

1. “Os estudos de caso visam à descoberta”, ou seja, mesmo partindo de alguns pressupostos teóricos o pesquisador se mantém atento às manifestações ao longo do estudo do caso em

questão. Essa característica pressupõe que o conhecimento não é algo acabado, mas uma construção;

2. “Os estudos de caso enfatizam a ‘interpretação em contexto’”, ou seja, situar o caso é fundamental para sua compreensão;

3. “Os estudos de caso buscam retratar a realidade de forma completa e profunda”;

4. “Os estudos de caso usam uma variedade de fontes de informação”, assim, os dados são coletados em diversos momentos, em situações variadas;

5. “Os estudos de caso revelam experiência vicária e permitem generalizações naturalísticas”, desse modo, o leitor poderá analisar o que pode ou não aplicar desse caso na sua situação;

6. “Estudos de caso procuram representar os diferentes e às vezes conflitantes pontos de vista presentes numa situação social”;

7. “Os relatos do estudo de caso utilizam linguagem e forma mais acessível”, tangenciando um estilo mais informal, buscando retratar de forma mais direta e clara como o caso ocorreu.

Diante do exposto e levando em consideração tais características de um estudo de caso, esclarecemos que a oficina de Pensamento Computacional é o nosso "caso" a ser estudado a partir de um propósito analítico. Concordando com Ponte (2006), acreditamos ser possível compreender a especificidade desse fenômeno, explorando-o de tal modo a sermos capazes de articular e discutir com nosso quadro teórico, propondo caminhos para o trabalho do PC com alunos da Educação Básica.

Ressaltamos ainda que, conforme Ponte (2006), é importante trazer o contexto em que o caso a ser estudado está inserido, visto que são os fatores internos e externos a esse caso que o definem como ele é. Há de se destacar que, para fazer um estudo de caso no contexto qualitativo, é necessária uma aproximação do pesquisador com seu objeto de estudo para que possa haver uma descrição mais fidedigna do objeto. André (2013, p.97) reforça tal fato, destacando que

> as abordagens qualitativas de pesquisa se fundamentam numa perspectiva que concebe o conhecimento como um processo socialmente construído pelos sujeitos nas suas interações cotidianas, enquanto atuam na realidade, transformando-a e sendo por ela transformados. Assim, o mundo do sujeito, os significados que atribui às suas experiências cotidianas, sua linguagem, suas produções culturais e suas formas de interações sociais constituem os núcleos centrais de preocupação dos pesquisadores. Se a visão de realidade é construída pelos sujeitos, nas interações sociais vivenciadas em seu ambiente de trabalho, de lazer, na família, torna-se fundamental uma aproximação do pesquisador a essas situações.

Sob tal perspectiva, sem a imersão do pesquisador no ambiente de sala de aula, discutir sobre a construção do Pensamento Computacional pelos estudantes, a interação dos alunos ao

longo do processo, é uma tarefa difícil de ser realizada. Dessa forma, a sala de aula se torna um ambiente de aprendizagem mútua, na medida em que pesquisador e alunos se transformam e se modificam durante o trabalho.

Nesse formato, estando imerso no ambiente da pesquisa e se transformando junto aos alunos, pesquisador também é sujeito da pesquisa, sendo o principal instrumento, na medida em que direciona o foco do trabalho, atribui significados ao fenômeno indagado, e ficando responsável por selecionar e coletar os dados, independente da sua natureza (BOGDAN e BIKLEN, 1999). Estar consciente dessa importância e de potencial influência na pesquisa, faz-se premente para que os devidos cuidados sejam tomados para que a pesquisa se torne mais fidedigna às manifestações espontâneas dos sujeitos.

Lüdke e André (2018) reforçam tal fato afirmando que, a crença de que havia uma perfeita separação entre o sujeito da pesquisa e o pesquisador, já caiu por terra. Conforme as autoras, o conhecimento não se faz de maneira imediata e transparente ao pesquisador, ao contrário, os dados vão sendo construídos a partir da interrogação e com base no que o pesquisador conhece sobre o assunto. O pesquisador, portanto, não é um sujeito de fora da pesquisa, mas está imbricado nos dados produzidos e, portanto, faz parte desse processo de construção do conhecimento a partir de suas crenças e convicções. Não há possibilidade de se separar pesquisador e o que ele estuda, seus resultados. Nas palavras das autoras, "é pelo seu trabalho como pesquisador que o conhecimento específico do assunto vai crescer, mas esse trabalho vem carregado e comprometido com todas as peculiaridades do pesquisador, inclusive e principalmente com as suas definições políticas" (LÜDKE; ANDRÉ, 2018, p.5).

Com essa consciência da responsabilidade do pesquisador no contexto de um estudo de caso no âmbito escolar é que ficamos imersos nessa oficina de Pensamento Computacional, objetivando ampliar nossa compreensão sobre esse trabalho.

3.2 Sobre a produção e coleta dos dados

Para além de classificar qual é o tipo de nossa pesquisa, um fator que merece relevante atenção em qualquer pesquisa científica é a coleta de seus dados. Buscando ser o mais fidedigno possível às manifestações dos sujeitos, procuramos produzi-los de maneira a garantir, ao máximo, a espontaneidade dos envolvidos, procurando representar de forma fidedigna o ambiente da sala de aula.

Conforme já apresentamos, nossos objetivos com a oficina de Pensamento Computacional no formato híbrido é proporcionar experiências plugadas (com a utilização de

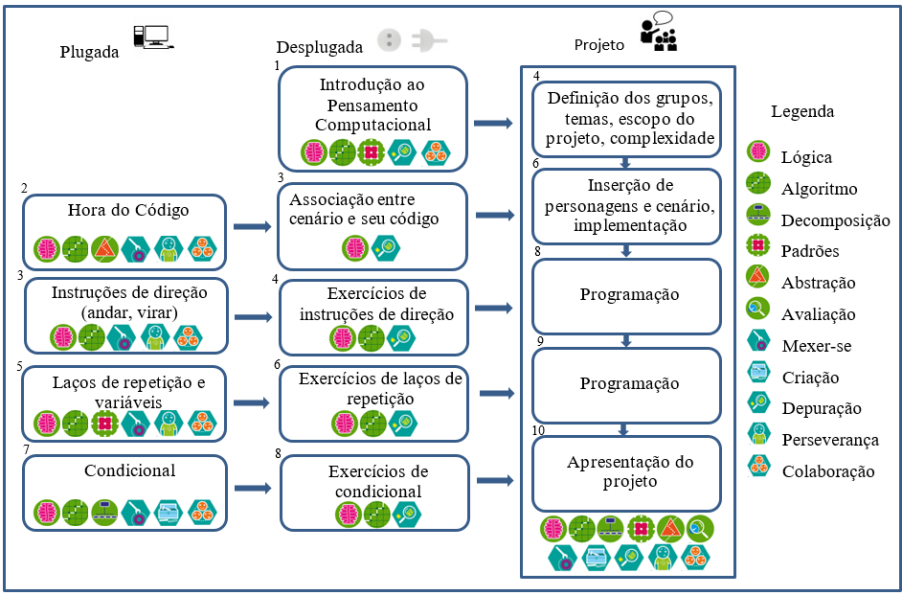
recursos digitais) e desplugadas (sem a utilização de recursos digitais) que explorem conceitos do PC, além da exploração desses conceitos, de forma mais aprofundada, por meio da programação em blocos. Esse formato híbrido para a oficina (alternância entre atividades plugadas, desplugadas), foi inspirado na metodologia para ensino do PC de Berto, Zaina e Sakata (2019) que propuseram a MEPeCoC (Metodologia para Ensino de Pensamento Computacional para Crianças).

Conforme as autoras, a metodologia proposta procura mesclar

> (i) atividades plugadas (que utilizam recursos digitais) para ensinar conceitos de pensamento computacional; (ii) atividades desplugadas (sem o uso de recursos digitais) para avaliar se o conceito foi aprendido e; (iii) desenvolvimento de um projeto (realizado em etapas) para relacionar e consolidar o aprendizado dos conceitos (BERTO; ZAINA; SAKATA, 2019, p.3).

A Figura 5 apresenta, de forma esquemática, a MEPeCoC que as autoras propuseram.

Figura 5 - Esquema da MEPeCoc



Fonte: Berto, Zaina e Sakata (2019, p.7)

A numeração ao lado de cada atividade indica a ordem de ocorrência, e os ícones constantes em cada momento apresentam as competências relacionadas ao PC que se pretende trabalhar. Conforme Berto, Zaina e Sakata (2019), esses momentos não necessariamente correspondem a uma aula, podendo ser também uma semana ou outro lapso temporal.

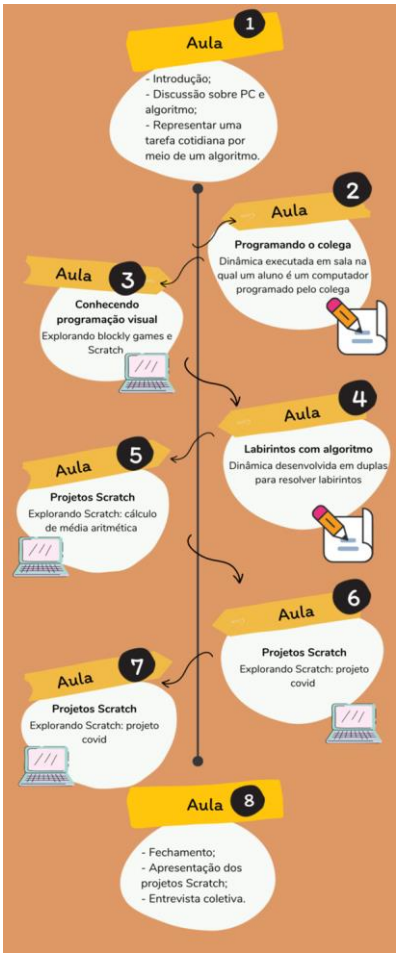
Destacamos que nossa proposta para construção dos dados da presente pesquisa se aproxima da MEPeCoC em alguns pontos e se distancia em outros. Na metodologia das autoras,

as atividades desplugadas são utilizadas para avaliar os conceitos ensinados via atividades plugadas. Em nossa proposta, tanto atividades plugadas e desplugadas serão utilizadas como forma de trabalhar os conceitos do PC de maneira gradual e incremental. Tanto na proposta das autoras quanto na que apresentamos, atividades plugadas e desplugadas são desenvolvidas até culminar no desenvolvimento de um projeto.

A produção de nossos dados se deu a partir de uma oficina sobre o Pensamento Computacional aplicada em 8 aulas de matemática de 50 min. Nosso público foram os 88 alunos das três turmas do sétimo ano do Ensino Fundamental do Colégio de Aplicação João XXIII da UFJF, na cidade de Juiz de Fora, MG. Esse trabalho contou com a colaboração do professor Leonardo Silva que prestou auxílio fundamental durante a oficina, oferendo suporte durante as atividades, especialmente na preparação da sala de informática. O desenvolvimento do trabalho em cada uma das turmas teve a duração de 2 semanas, sendo 4 aulas por semana.

Conforme já comentado, as atividades desenvolvidas procuraram alternar entre plugadas, desplugadas e desenvolvimento do projeto, conforme ilustrado na Figura 6:

Figura 6 - Cronograma de aula da oficina



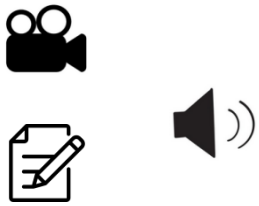
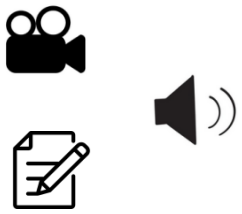
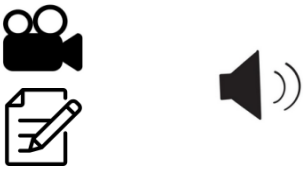
Fonte: o autor

A aplicação da oficina ocorreu inicialmente nas turmas B e C. Após a conclusão da aplicação nessas turmas, pequenos ajustes foram feitos e então foi aplicada na turma A. Os detalhes da aplicação em cada uma das turmas serão apresentados em um capítulo exclusivo.

Durante a aplicação da oficina, registramos os dados por meio de quatro formatos: áudios (gravados via aplicativo de comunicação instantânea pelo celular do pesquisador), vídeos (gravados por meio de uma filmadora colocada dentro da sala de aula), registros em papel (feitos pelos próprios alunos ao desenvolverem as atividades) e, vídeos de captura da tela do computador (gravados por *software* do computador durante o desenvolvimento do projeto).

O Quadro *2* ilustra como foram gerados os dados em cada um dos encontros da oficina.

Quadro 2 - Formato de produção dos dados

	O que aconteceu nesse encontro? / Pilares do PC e conceitos computacionais que foram explorados / distribuição dos alunos em cada aula	**Como foram gerados dados nesse encontro**
Aula 1	- Introdução ao trabalho; - Discussão sobre o PC e algoritmo; - Descrição de uma tarefa cotidiana por meio de algoritmo. Pilares: decomposição, abstração e algoritmo; Conceitos computacionais: sequências e condicionais. Distribuição dos alunos: atividades individuais	
Aula 2	Atividade desplugada: computador e programador. Pilares: decomposição, reconhecimento de padrões, abstração e algoritmo; Conceitos computacionais: sequências. Distribuição dos alunos: atividades em duplas	
Aula 3	Atividade plugada: trabalhando com Blockly Games (turmas B e C) e Scratch (turma A). Pilares: decomposição, reconhecimento de padrões e algoritmo; Conceitos computacionais: sequências, loops, eventos e condicionais. Distribuição dos alunos: atividades em duplas	
Aula 4	Atividade desplugada: labirintos. Pilares: decomposição, reconhecimento de padrões, abstração e algoritmo; Conceitos computacionais: sequências e loops. Distribuição dos alunos: atividades em duplas	
Aulas 5, 6 e 7	Atividade plugada: trabalhando projetos no Scratch. Pilares: decomposição, reconhecimento de padrões, abstração e algoritmo;	

	Conceitos computacionais: sequências, loops, eventos, paralelismo, condicionais, operadores e dados. Distribuição dos alunos: atividades em grupos/duplas	
Aula 8	Finalização: apresentação dos projetos, discussões e entrevista coletiva.	

Fonte: o autor

LEGENDA

 gravação em vídeo com áudio	 gravação em áudio via app de comunicação no celular do pesquisador	 anotações feitas em papel pelos alunos (desenvolvimento das atividades desplugadas e anotação sobre projeto Scratch)	 captura de tela do computador durante desenvolvimento de projetos no scratch com áudio

Destacamos que, no primeiro dia da oficina, trabalhamos com os alunos individualmente, debatendo sobre os algoritmos e depois solicitando que pensassem em um problema de seu cotidiano, escrevendo no papel a solução para esse problema. Dessa forma permitimos que todos tivessem a oportunidade de refletir sobre seu cotidiano e compreenderem como o PC está conectado em seu dia a dia.

Para os demais dias de aulas desplugadas, o próprio formato da oficina ensejou o trabalho em duplas. Na aula 2, cada aluno desempenhava um papel na atividade (programador/computador) e, na aula 4, trabalhavam em conjunto para solucionar o problema proposto.

Já nas aulas plugadas, utilizamos o formato em duplas devido a limitação de computadores. Uma exceção ocorrida foi em relação ao desenvolvimento do projeto maior no Scratch que, na primeira turma, fizemos em equipes de 5 alunos, objetivando que discutissem sobre o problema para avançarem, porém, percebemos que vários não estavam se envolvendo no trabalho e, portanto, para as demais turmas, voltamos para o formato de duplas, o que ocasionou um maior envolvimento dos alunos e debate entre eles para solucionarem os problemas.

Para além do formato de distribuição dos alunos ao longo da oficina, destacamos que o volume de dados gerados permitiu analisarmos criteriosamente o seu processo de desenvolvimento, o que abriu oportunidade para refletir sobre o papel que cada atividade,

plugada ou desplugada, desempenhou como potencializadora, ou não, do PC, sendo possível tecer discussões que transcenderam o Pensamento Computacional.

3.3 Sobre os sujeitos da pesquisa

Os estudantes participantes das atividades são os alunos das três turmas do sétimo ano do Ensino Fundamental do Colégio de Aplicação João XXIII da Universidade Federal de Juiz de Fora. O rol de alunos é composto por 88 estudantes, sendo 30 da turma 7A, 28 do 7B e 30 do 7C, todos com idade próxima de 12 anos.

Destacamos que o formato de ingresso dos estudantes no Colégio de Aplicação se dá por meio de sorteio público, ou seja, as turmas são heterogêneas com alunos oriundos de diferentes realidades sociais.

Todos esses alunos foram convidados previamente a participarem do trabalho, e, na medida que aceitaram o convite, assinaram o Termo de Assentimento Livre e Esclarecido (TALE) e seus responsáveis assinaram o Termo de Consentimento Livre e Esclarecido (TCLE). A presente pesquisa está devidamente registrada no comitê de ética da Fiocruz, sob o parecer de número 4.821.756.

3.4 Sobre a forma de discussão e avaliação do PC

Conforme observado a partir da literatura, a maior parte das intervenções relacionadas ao PC busca avaliar esse tipo de pensamento por meio de quantificadores, testes, notas. Essas avaliações quantitativas, muitas vezes focam no produto (que normalmente são códigos de programação), ou focam no resultado da aplicação de testes. Em nossa discussão, procuraremos nos distanciar desse formato avaliativo, entendendo que ele se mostra limitado sob nossa ótica do PC.

Acreditamos que, ao nos voltarmos com olhar atento à oficina de PC desenvolvida com os alunos do Ensino Fundamental II, teremos a oportunidade de discutir o processo de construção de conhecimento dos alunos a partir da observação dos detalhes envolvidos, de manipulações de conceitos computacionais e de experiências múltiplas vividas pelos alunos a partir da abordagem plugada e desplugada. Nesse formato, estaremos abertos a questões que porventura surjam no desenrolar dessa oficina que estejam além do PC, ampliando nosso leque de discussão de questões que podem surgir a partir de uma oficina de Pensamento Computacional.

Ao analisarmos os dados produzidos a partir dessa oficina (áudios, anotações/atividades, vídeos, projetos Scratch), observaremos como os conceitos computacionais propostos por Brennam e Resnick (2012) - Sequência, Loops, Eventos, Paralelismo, Condicionais, Operadores e Dados estarão sendo manipulados e tangenciados; observaremos também como os quatro pilares do PC – Decomposição de problemas, Abstração, Reconhecimento de Padrões e Algoritmos (LIUKAS, 2015; BBC BITSIZE, 2015; BRACKMANN, 2017) estarão sendo explorados e manipulados pelos alunos ao longo das atividades e, com isso, buscaremos observar algum nível de evolução dos alunos e se houve indícios de que o PC foi potencializado a partir de nossa abordagem. Ao longo da análise dos dados esses termos ficarão destacados em itálico para facilitar nossa discussão e entendimento em relação ao tema proposto para nossa pesquisa.

3.5 Síntese da metodologia de pesquisa

Procurando sintetizar os procedimentos metodológicos da presente pesquisa, construímos na Figura *7* um pequeno infográfico para indicar seus principais pontos.

Figura 7 - Metodologia da pesquisa



Fonte: o autor

A imagem aponta, de maneira simplificada, os detalhes que perpassam a metodologia da pesquisa. O detalhamento referente a aplicação da oficina que gerou os dados da pesquisa será apresentado no capítulo seguinte.

4 SOBRE A INTERVENÇÃO

Inicialmente a intervenção foi planejada para ser desenvolvida com duas turmas de sétimo ano do Ensino Fundamental (7B e 7C) devido à proximidade e acesso à professora de Matemática dessas turmas pelo pesquisador, porém, ao estar na escola e conhecer a professora da outra turma (7A), tivemos abertura para também realizar a intervenção com esses alunos. Dessa forma, devido ao choque de horários, inicialmente foi feita a intervenção nas turmas 7B e 7C ao longo de duas semanas e, após esse período, por mais duas semanas, procedemos ao desenvolvimento das atividades com a turma 7A.

Esse formato, inicialmente não planejado, favoreceu olhar para a primeira aplicação nas turmas 7B e 7C, fazer algumas reflexões e ajustes de detalhes na sequência de atividades, para então proceder à aplicação na turma 7A. Os pormenores e algumas reflexões iniciais sobre a intervenção são apresentados na sequência.

Nesse capítulo apresentaremos de forma exaustiva e detalhada todo o processo de aplicação da oficina nas turmas, trazendo diversas ilustrações das atividades, transcrições de áudios e pequenas discussões sobre o processo. Entendemos que dessa forma poderemos trazer o leitor para dentro da sala de aula conosco, vivenciando os mais diversos momentos que experienciamos ao longo de todas as aplicações.

4.1 Primeira aplicação da sequência didática

Destacamos que essa primeira aplicação aconteceu em duas turmas do sétimo ano do Ensino Fundamental (7B e 7C), sendo uma turma com 30 alunos e outra com 28 alunos. Com objetivo de deixar a leitura mais fluida, iremos relatar primeiramente cada um dos encontros com a turma 7B e posteriormente os encontros com a turma 7C.

Como a intervenção aconteceu de forma muito semelhante em ambas as turmas, apresentaremos com mais detalhes a turma 7B e, para a turma 7C, daremos mais ênfase ao que diferiu da primeira turma.

4.1.1 7B

Segundo a professora de Matemática, o 7B é uma turma desafiadora. Conforme relatou, é sempre necessário estar chamando atenção dos alunos por serem bastante dispersos. Essa turma é composta por 30 alunos com idades próximas de 12 anos.

4.1.1.1 Aula 1 (21/09/2022) – Introdução e atividade desplugada (10:50 às 11:40)

Foi iniciado um debate com os alunos sobre o que seria o PC. Nesse bate papo inicial, os alunos foram apresentando suas primeiras impressões sobre o assunto. Eles relataram se tratar de um tipo de pensamento matemático; pensamento na tecnologia; outro disse estar ligado à programação de computadores ou à lógica matemática.

A partir dessas primeiras impressões dos alunos sobre o assunto, foi sendo construído um debate de que o PC está relacionado principalmente à matemática no que diz respeito à resolução de problemas. Assim, concluímos que o PC está mais ligado a resolver problemas de maneira que a solução seja dada por uma sequência de passos que qualquer um consiga compreender, seja um humano ou não.

Inspirados em Brackmann (2017), foi colocado na lousa a soma 137 + 281, com um número embaixo do outro. Nesse momento, foi solicitado que os alunos "ensinassem" ao pesquisador a fazer essa soma. A princípio os alunos foram logo querendo dizer o resultado, porém, foi destacado que, o que gostaria de saber, é como fazer a conta, e não o resultado em si. Nessa discussão surgiu a palavra "algoritmo" dita por um dos alunos. Nesse momento foi questionado à turma o que seria o algoritmo e um aluno descreveu que

um algoritmo é tipo um passo a passo que tem que seguir, por exemplo, atravessar a rua. Espera o sinal ficar verde... atravesse a rua. (SIC) Aula 1 - 7B - 3:33 - 3:43

Assim, buscando conexão com a fala do aluno, chegamos à conclusão de que um algoritmo seria uma sequência de passos a serem seguidos para resolver um problema.

A partir dessa discussão, retomamos a questão da soma dos números e os alunos foram dizendo a sequência de passos (agora já o entendendo como um algoritmo) para resolvê-la, ou seja, somar primeiro as unidades, depois as dezenas e então as centenas. Ao fazer isso no quadro, chegamos ao valor 3118. Os alunos logo se manifestaram dizendo não ser dessa forma a soma, foi então que um novo questionamento surgiu, destacando que eu (pesquisador) era um robô que estava apenas executando o que eles haviam me ensinado. Foi então que discutimos sobre uma "condicional/verificação" a ser acrescentada, visto que, quando essa soma fosse igual ou superior a 10, o valor da dezena gerada deveria ser "passada para o lado" e então dar sequência nessa soma. Após essa discussão foi concluído o cálculo e ilustrado que o algoritmo se trata de uma sequência de passos ordenados a serem dados, de forma que qualquer um possa

compreendê-los e executá-los, ressaltando que é preciso de alguma forma “prever” possíveis falhas na interpretação, como ocorreu com a soma.

Como nova ilustração e meio para ensejar mais debate sobre o assunto, foi apresentado um pequeno vídeo[12] de um robô aspirador executando sua tarefa. A partir do vídeo, abrimos uma discussão sobre “qual era o problema do robô” – limpar – e qual seria seu algoritmo para resolver esse problema. Começamos a debater sobre uma possível programação que estava embutida no robô para desempenhar tal empreitada. Comentamos sobre os obstáculos que ele encontra e como se porta diante deles (ao tocar em um obstáculo dá uma pequena ré e gira 180 graus e segue em seu trabalho).

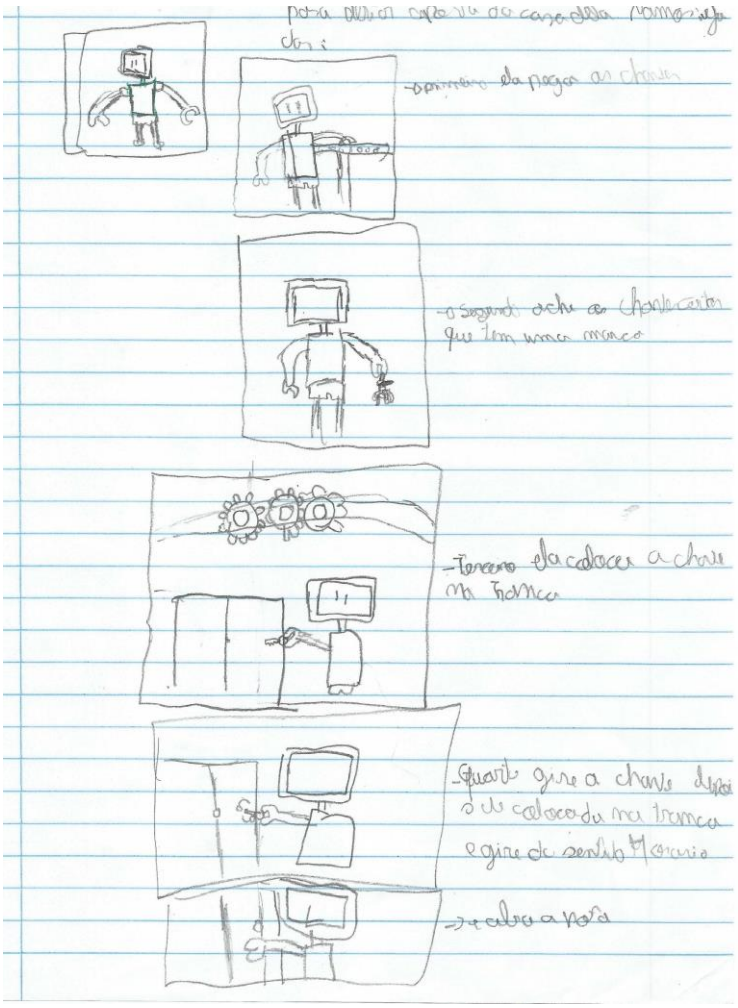
Após essas discussões iniciais e primeiros entendimentos do que é um algoritmo, foi solicitado que os alunos pensassem em alguma tarefa cotidiana e descrevessem-na como se estivessem “ensinando” um robô a desempenhar essa função. O objetivo dessa atividade era que os alunos fossem capazes de construir uma sequência de passos ordenados para executar determinada tarefa, considerando possíveis "problemas" de interpretação que o robô pudesse ter, tentando antecipar essas questões e colocando algumas condicionais em sua sequência de passos.

Antes dos alunos iniciarem a atividade, como exemplo, foi dito que iria ensinar um robô a “escovar os dentes”. Assim, ficou claro aos alunos que esse era o problema do robô e que para solucioná-lo era preciso dizer a ele, com detalhes, cada passo a ser dado. Seguindo o exemplo, foi detalhado que primeiro o robô deveria ir ao banheiro, depois abrir a segunda gaveta do armário, pegar a escova e pasta, abrir a pasta, colocar um pouco de pasta na escova e assim por diante, até concluir a tarefa de “escovar os dentes”.

Terminada essa pequena explicação, os alunos, em uma folha do caderno, descreveram sua “programação” do robô para desempenhar uma tarefa cotidiana qualquer, por eles determinada. A Figura 8 e a Figura 9 a seguir ilustram a descrição feita por dois dos alunos dessa turma:

[12] Esse vídeo que apresenta um robô aspirador em funcionamento foi criado pelo pesquisador em sua casa, porém, a ideia de apresentá-lo como fonte de discussão do PC foi inspirada na aula de pensamento computacional da professora Maria da Graça (UNIVESP, 2022).

Figura 8 - Atividade 1 aluno A

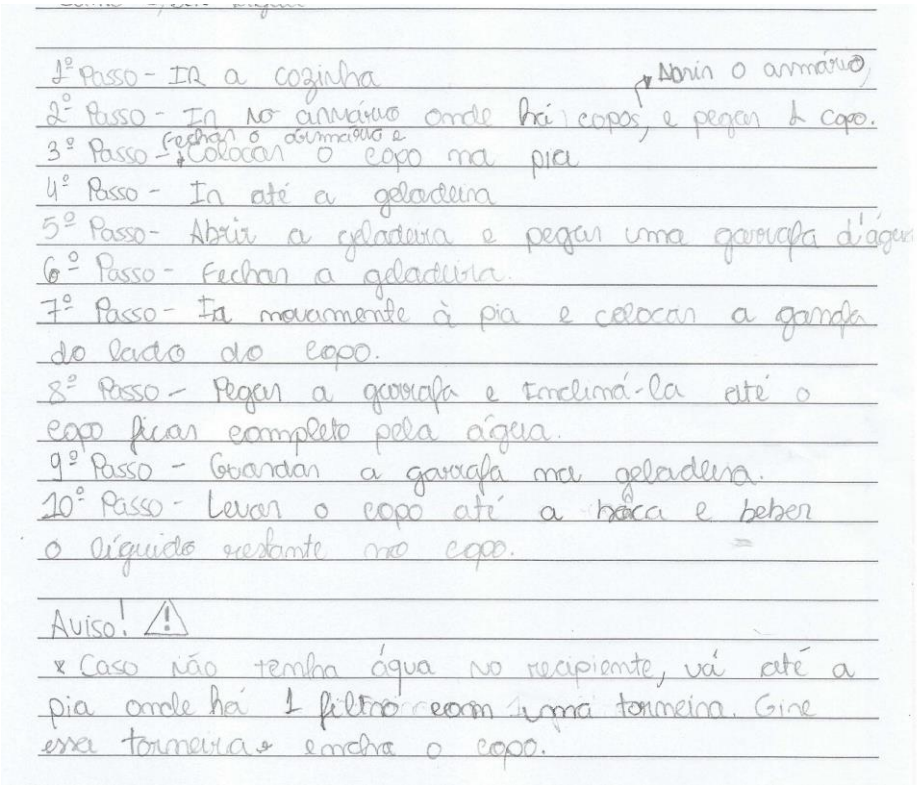


Fonte: o autor

Nessa ilustração, é possível observar que o aluno desenha um robô nomeado de Karen que tem o problema de abrir a porta de casa. Na sequência, com ilustrações, o aluno vai dando os passos a serem seguidos por Karen para executar essa tarefa (pegar as chaves; achar a chave correta que tem uma marca; colocar a chave na tranca; girar a chave depois de colocada na tranca no sentido horário; abra a porta).

Já o outro aluno cria uma sequência de passos para "beber água", conforme é possível visualizar na Figura 9.

Figura 9 - Atividade 1 aluno B



Fonte: o autor

É possível observar, no final da sequência de passos, que o aluno inseriu um alerta, podendo ser interpretado como uma condicional, dizendo que, caso não haja água no primeiro local indicado, seria necessário se dirigir a outro local. É importante ressaltar que esse "alerta" muitas vezes foi deixado de lado pelos alunos, porém, o pesquisador foi incentivando, por meio de problematizações, os alunos a inserirem essa questão condicional, objetivando forçá-los a refletirem sobre a resolução desse problema e possíveis "falhas" que poderiam ocorrer caso algo não saísse exatamente como o esperado.

A seguir está transcrito um diálogo entre o pesquisador e uma aluna. Essa aluna chamou o pesquisador para mostrar sua sequência de passos já escrita para ensinar um robô a tomar banho. O pesquisador pega a folha da aluna e, ao seu lado, e começa a ler em voz alta.

Pesquisador: Vamo lá. Ir para o banheiro; tirar a roupa; entrar no box; ligar o chuveiro; molhar o corpo e o cabelo; pegar o shampoo, abrir, coloca na mão, passar no cabelo, tirar o shampoo do cabelo; pegar o sabonete... é isso aí. Vamo dá uma complicada agora. A hora que eu fui colocar o shampoo, tinha acabado o shampoo. O que que você pode acrescentar puxando uma setinha de condicional, tipo nessa soma aqui que vocês me ensinaram

[pesquisador aponta para o quadro], passou de 10, eu tive que fazer uma coisa, por exemplo, na hora que eu fui usar o shampoo, não tinha shampoo. Se o shampoo acabou...

Aluna: Ver no armário...

Pesquisador: Beleza

Vários alunos apresentaram suas sequências de passos e, sempre que possível, eram instigados a pensar mais sobre seus problemas a partir de problematizações feitas pelo pesquisador.

Ao final do encontro foram feitas algumas discussões com os alunos sobre as sequências de passos por eles construídas e suas anotações foram recolhidas encerrando assim a primeira aula.

Ao longo dessa primeira aula, os alunos puderam trabalhar um pouco do Pensamento Computacional como um auxiliador na resolução de problemas. Desde as discussões iniciais até a finalização da aula com a atividade desenvolvida pelos alunos, pudemos explorar um pouco dos pilares do PC: Decomposição de problemas, Abstração e Algoritmo, além dos conceitos computacionais de: Sequência, Eventos e Condicionais. Mais detalhes em relação a essa exploração serão apresentados no capítulo referente a análise dos dados.

4.1.1.2 Aula 2 (22/09/2022) – Atividade desplugada (8:55 às 9:40)

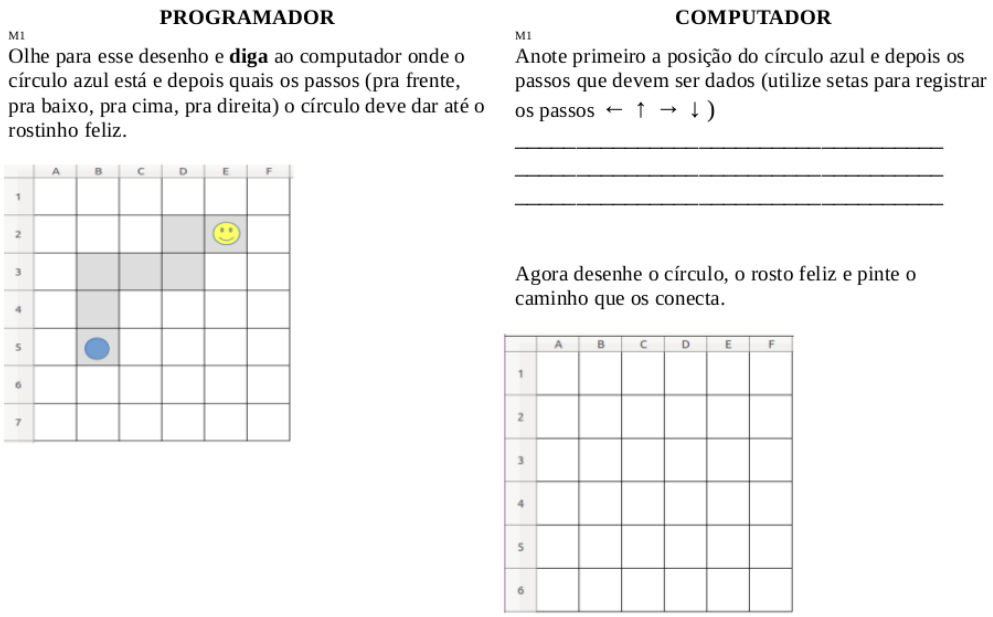
Inicialmente destacamos que esse horário de aula é menor, de apenas 45min. Essa mudança na duração da aula se deu logo após a pandemia da COVID-19, devido a ajustes necessários para o retorno às aulas presenciais.

Nesse dia aconteceu a segunda sequência de atividades desplugadas por meio de uma dinâmica em duplas. Para iniciar o trabalho foi solicitado aos alunos que organizassem as carteiras de modo que a sala ficasse composta por 3 fileiras de duplas.

A proposta consistia em trabalhar com os alunos de maneira que um deles fosse o programador e o outro o computador, assim, conforme havíamos discutido na aula anterior, o computador (robô) não sabe nada, apenas executa os comandos exatamente como o programador mandar.

Cada dupla recebeu um pedaço de folha de papel que continha um quadriculado em branco (computador) e o outro recebeu um com quadriculado contendo um percurso desenhado (programador). Fazendo referência à aula anterior, foi dito que o programador deveria dizer (programar) ao computador a localização dos objetos no quadriculado e indicar o caminho a ser percorrido pela bolinha até o rosto feliz. A Figura 10 a seguir ilustra os enunciados recebidos pelos alunos na primeira parte da dinâmica.

Figura 10 - Enunciado 1 atividade desplugada

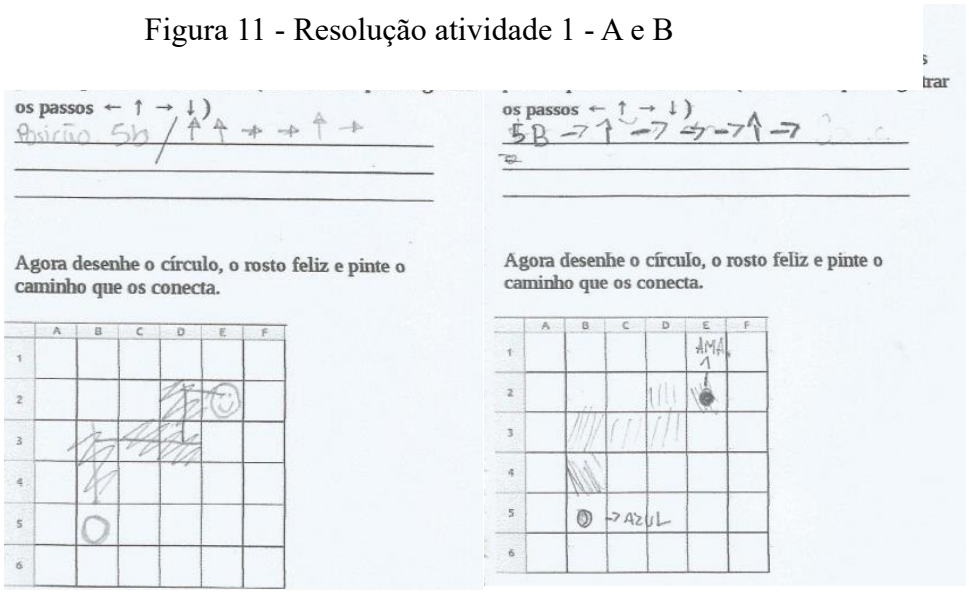


Fonte: o autor

A princípio, quando a atividade foi elaborada, esperava-se que os alunos dissessem a quantidade de passos a serem dados (2 para a direita, 3 para cima, por exemplo), e então o computador fosse escrevendo as setas conforme tais comandos, porém, vários utilizaram outro recurso dizendo “ande para a direita até chegar na casa C5”, por exemplo, ou então dizendo cada uma das casas que deveria ir passando para alcançar o objetivo.

Ilustramos algumas resoluções feitas pelos alunos a seguir.

Figura 11 - Resolução atividade 1 - A e B

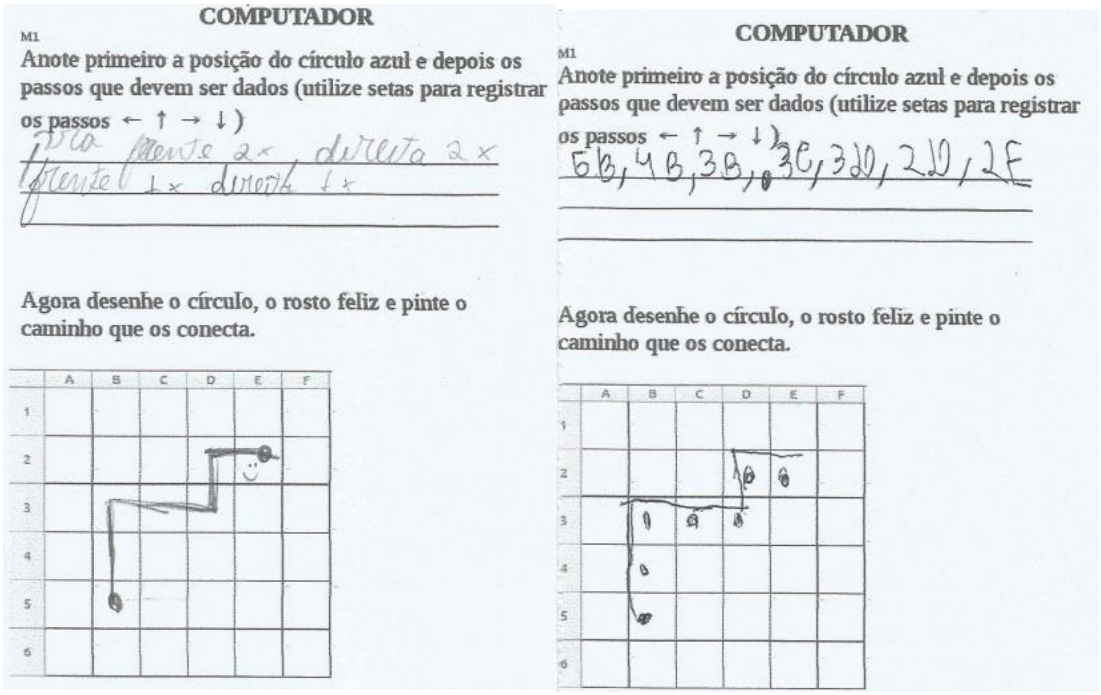


Fonte: o autor

As resoluções apresentadas são das duplas 5 e 11, respectivamente. Era esperado que todos os alunos fizessem algo parecido, anotando primeiramente a localização do círculo azul, depois o percurso a ser seguido utilizando setas e, por fim, construíssem o caminho no quadriculado (conforme orientação do enunciado). Das 14 duplas que estavam presentes nessa aula, apenas 5 utilizaram as setas para indicar o caminho e dessas, apenas 3 anotaram a posição inicial do círculo azul.

As outras 9 duplas utilizaram estratégias distintas para resolver esse problema conforme exemplificamos a seguir.

Figura 12 - Resolução atividade 1 - C e D



Fonte: o autor

As soluções apresentadas são das duplas 1 e 2, respectivamente. A primeira dupla escreveu a quantidade de vezes que a bolinha deveria andar para cada direção, enquanto a segunda dupla foi escrevendo cada um dos quadros (como células de uma planilha eletrônica) que a bolinha azul deveria ir percorrendo.

Ambos resolveram o problema e conseguiram atingir o objetivo proposto, porém, é possível perceber que não se atentaram ao enunciado da atividade que solicitava a utilização de setas para indicar o percurso. Tal fato merece destaque visto que foi possível observar a dificuldade de leitura dos alunos não só nesse momento, mas durante a execução da atividade. Foi preciso explicar para várias duplas o que deveria ser feito, visto que os alunos solicitavam atendimento dizendo que não haviam entendido.

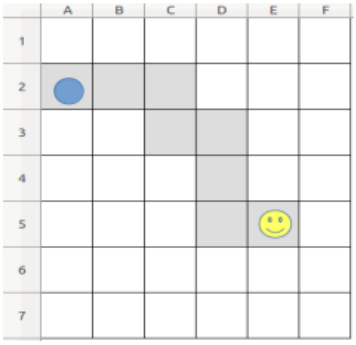
No segundo momento a atividade foi repetida (guiar a bolinha até o rosto feliz), porém, com um caminho diferente, conforme ilustrado na Figura 13.

Figura 13 - Enunciado atividade 2 desplugada

PROGRAMADOR

M2

Olhe para esse desenho e **diga** ao computador onde o círculo azul está e depois quais os passos (pra frente, pra baixo, pra cima, pra direita) o círculo deve dar até o rostinho feliz.



COMPUTADOR

M2

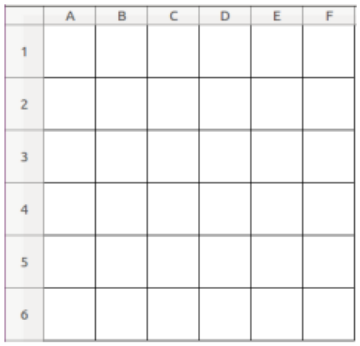
Anote primeiro a posição do círculo azul e depois os passos que devem ser dados (utilize setas para registrar os passos ← ↑ → ↓)

_____

_____

_____

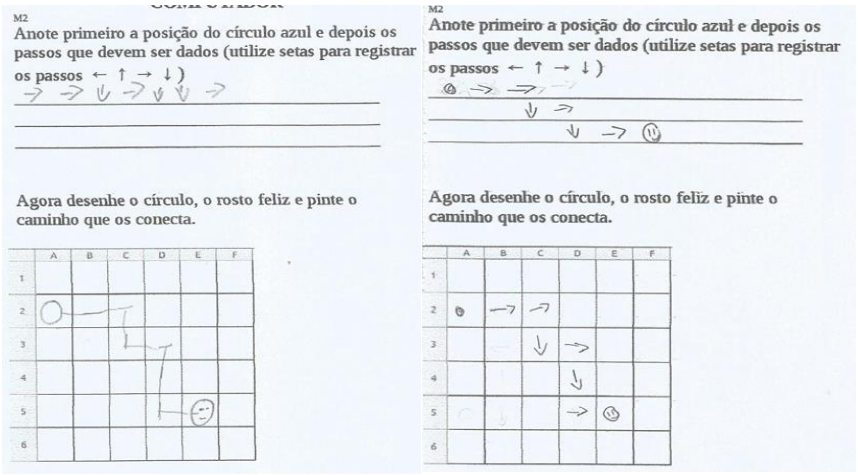
Agora desenhe o círculo, o rosto feliz e pinte o caminho que os conecta.



Fonte: o autor

Destacamos que para esse segundo momento os papéis dos alunos dentro de suas duplas foram trocados, ou seja, aquele que era computador virou o programador e vice-versa. A Figura 14 apresenta a resolução das duplas 5 e 11 novamente.

Figura 14 - Resolução atividade 2 - A e B

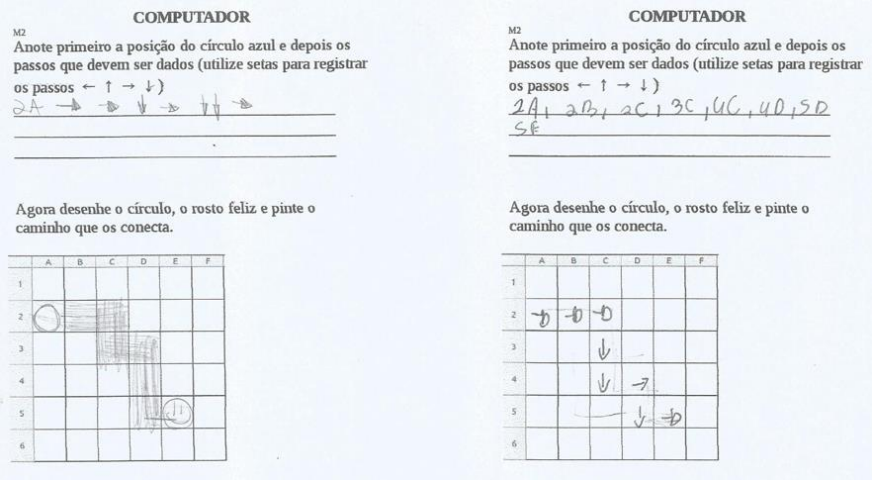


Fonte: o autor

É interessante observar que para o segundo momento, nenhuma das duas duplas anotou a posição inicial da bolinha conforme foi feito no primeiro momento da atividade (papéis invertidos).

Já na resolução das duplas 1 e 2 é possível observar que a dupla 1 anotou a posição inicial do círculo e depois utilizou as setinhas para traçar o caminho a ser percorrido, exatamente como era esperado que todos fizessem. A resolução dessas duplas está ilustrada a seguir pela Figura *15*:

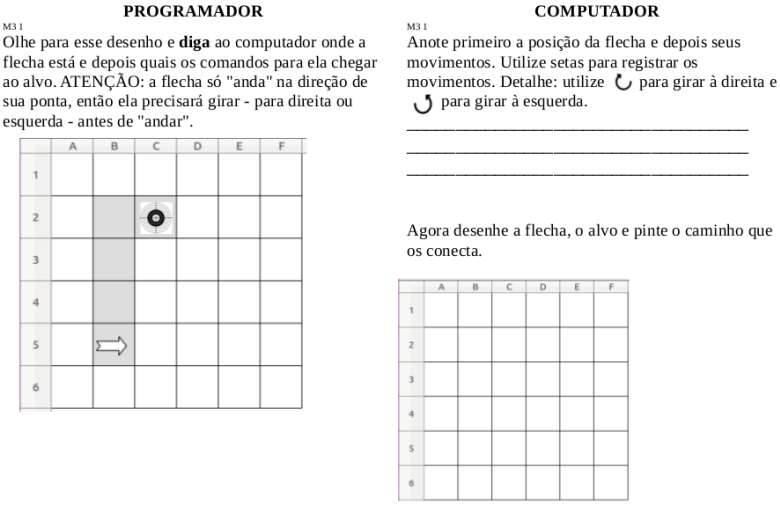
Figura 15 - Resolução atividade 2 C e D



Fonte: o autor

Terminadas essas etapas, o momento seguinte se deu forma semelhante, porém, agora seria necessário utilizar também um recurso de “curva”. Eles receberam um quadriculado com um caminho, porém, agora, deveriam guiar uma flecha até um alvo. Assim, segundo instrução, a flecha só poderia se movimentar para frente, na direção de sua ponta, sendo necessária a utilização dos comandos de giro (para a direita ou esquerda). A atividade está ilustrada na sequência.

Figura 16 - Enunciado atividade 3 desplugada

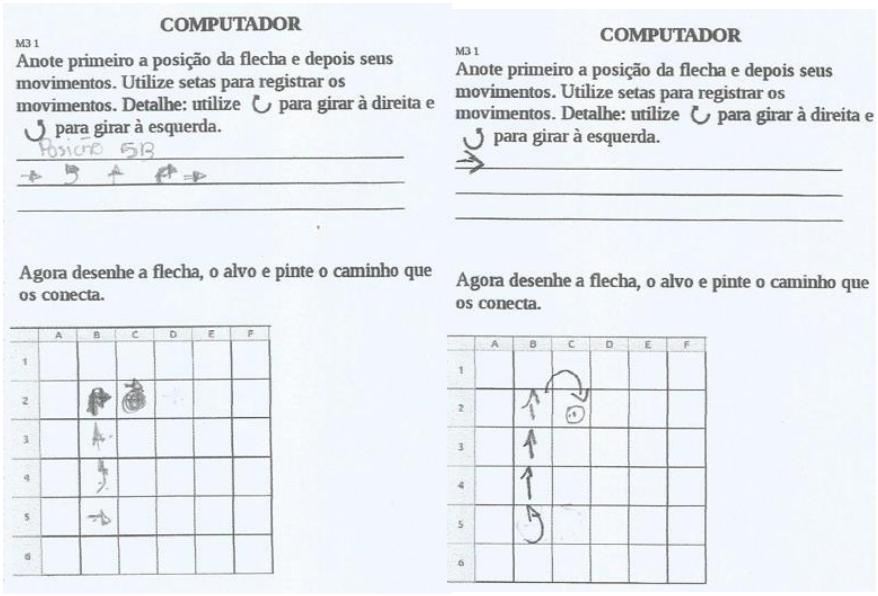


Fonte: o autor

A utilização do recurso de giro gerou muita confusão, especialmente porque os alunos, ao invés de dizerem isso, ficaram mostrando com a mão para o colega qual o lado deveria girar.

A seguir ilustramos as resoluções feitas pelas duplas 5 e 11, respectivamente.

Figura 17 - Resolução atividade 3 - A e B

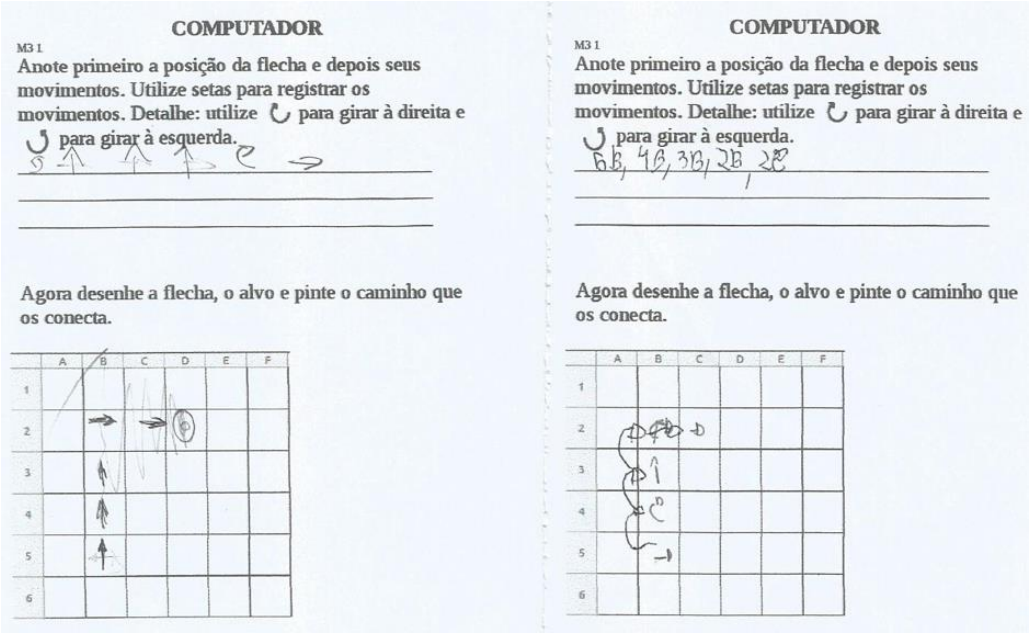


Fonte: o autor

É possível observar que a questão do giro trouxe muita confusão para a representação por parte dos alunos. Eles não conseguiam visualizar, por exemplo, que a setinha com curva apenas girava a flecha no mesmo local que ela estava, sem fazer com que ela movesse para frente.

Quando começaram a entender a dinâmica do giro, foi possível perceber ainda que, mesmo com a flecha já apontada para a direção que precisava, eles davam o comando de “andar para cima” ao invés de “andar para frente”, visto que a flecha realmente precisava ir para cima, porém, como já estava apontando para cima, era preciso dizer que ela fosse para frente. Tal fato é possível observar na resolução da dupla 1 ilustrada a seguir. Já a dupla 2, também na ilustração abaixo, utiliza o recurso de ir anotando cada uma das células que a flecha deveria percorrer e, o giro, era colocado apenas no quadriculado que deve constar o percurso.

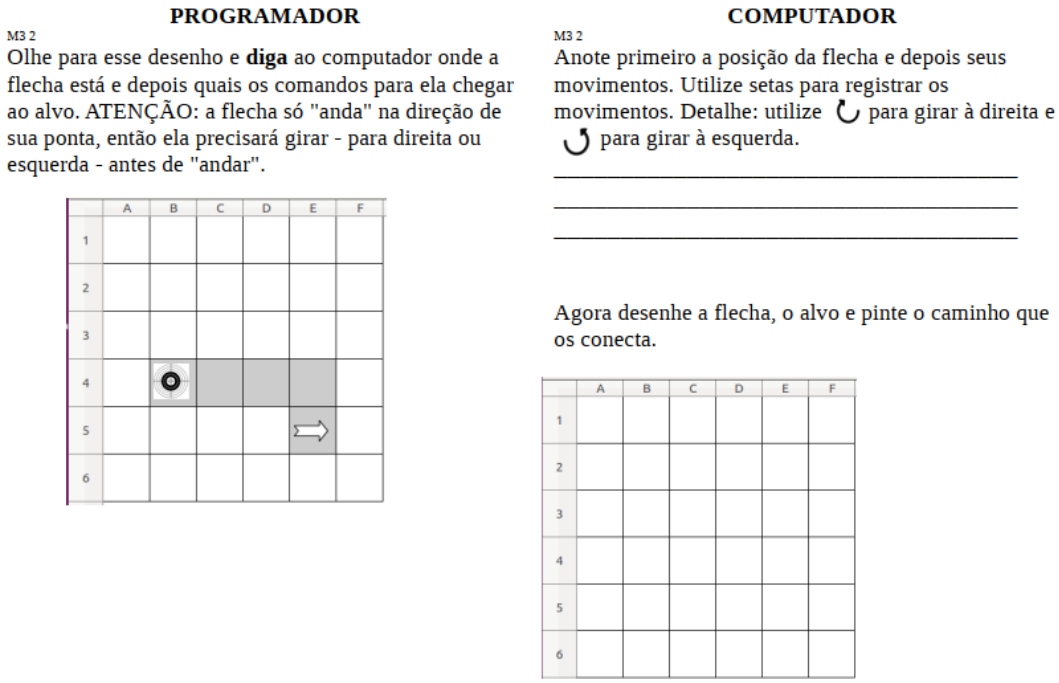
Figura 18 - Resolução atividade 3 - C e D



Fonte: o autor

Além disso, outra questão que gerou grandes discussões entre os alunos, foi quando pediam ao colega para andar para direita ou esquerda, eles diziam “minha direita ou a sua?”. Essa dúvida surgiu, pois estavam um de frente para o outro, fazendo com que interpretassem que os movimentos deveriam ser espelhados. Muitos não estavam compreendendo que, ao dizer “andar para direita” seria o mesmo para os dois, visto que ambos estavam olhando para o papel quadriculado. Isso gerou vários debates entre os alunos. A dinâmica da flecha aconteceu novamente com outro caminho e agora com os alunos trocando de papéis. O novo percurso a ser percorrido pela flecha está ilustrado na Figura *19*:

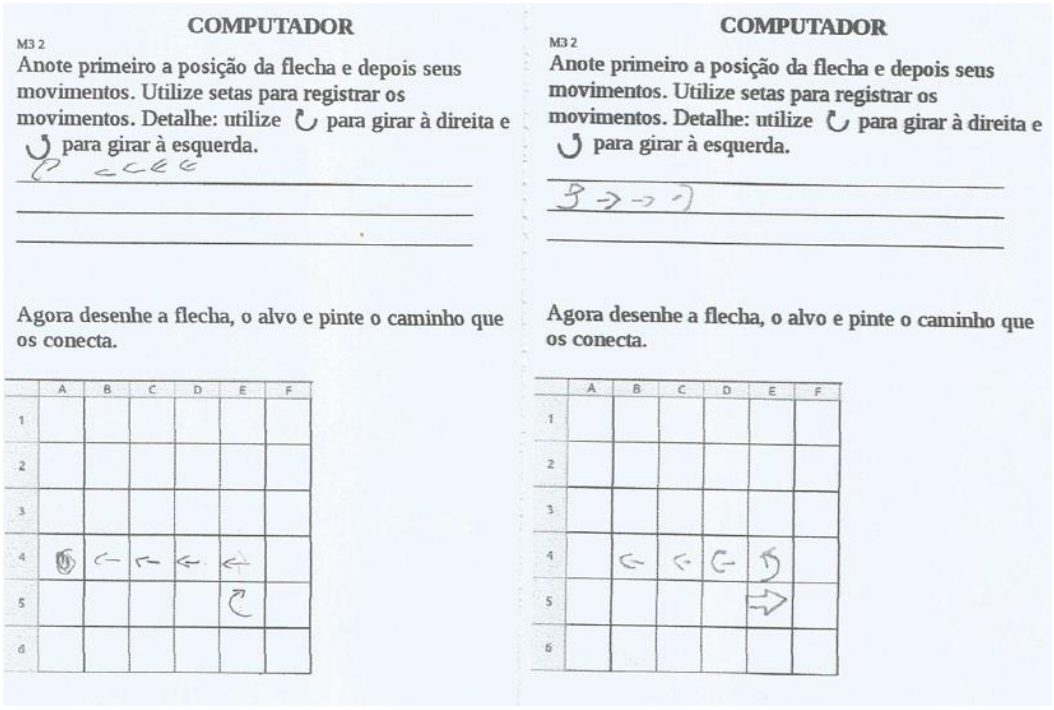
Figura 19 - Enunciado atividade 4



Fonte: o autor

Para a resolução dessa etapa tanto a dupla 5 quanto a dupla 11 se confundiram em relação à utilização do comando de giro, visto que, pela posição da flecha, era preciso iniciar com um giro, depois um passo, outro giro, para só então se movimentar até o alvo. Conforme podemos observar na ilustração, tanto a dupla 5 quanto a dupla 11 utilizam apenas um único giro logo de início e depois os passos até o alvo, subentendendo-se que ambos consideraram o comando de giro como já o primeiro movimento da flecha para outro quadrado.

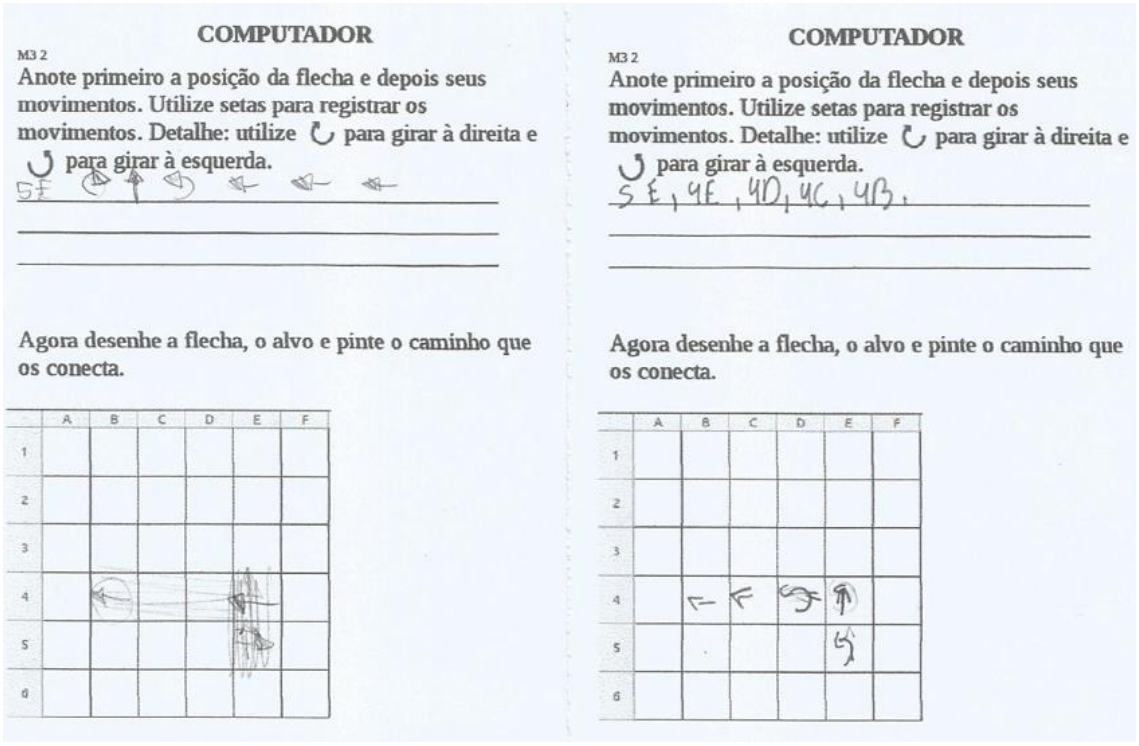
Figura 20 - Resolução atividade 4 - A e B



Fonte: o autor

Já na resolução da dupla 1 é possível observar que houve apenas uma confusão entre os giros para a direita e para a esquerda, porém, os demais passos atendem ao percurso traçado. Já a dupla 2 parece ter entendido qual o percurso a ser feito, porém, não utiliza as setas para indicar o caminho, mas sim o “nome” das células, desenhando as setas onde deveria estar apenas o caminho pintado.

Figura 21 - Resolução atividade 4 - C e D



Fonte: o autor

Com esse último momento encerramos o segundo dia de trabalho tendo oportunizado aos alunos uma experiência de como construir um algoritmo utilizando apenas setas para guiar um objeto até determinado alvo. À primeira vista parecia uma atividade relativamente simples, porém, oportunizou diversas discussões que tangenciaram lateralidade, giros e formas de representação dos movimentos.

É importante destacar que algumas duplas estavam mais desinteressadas pelo trabalho e brincaram muito durante a dinâmica, incomodando outros colegas que estavam engajados na atividade. Outros ainda estavam muito preocupados em “acertar” o caminho e, foi possível observar que estavam olhando o percurso pronto que estava com seu colega, como se fosse um “gabarito” para corrigir o que haviam feito antes de entregar o trabalho.

Tal fato não causa tanta estranheza visto que estão acostumados ao padrão tradicional de ensino no qual são sempre avaliados, tendo seus acertos nas atividades transformados em notas. Mesmo tendo sido deixado claro no início que se tratava de uma atividade voluntária e que não "valia ponto", havia a preocupação em acertar.

4.1.1.3 Aula 3 (22/09/2022) – Atividade plugada (10:50 às 11:40)

O objetivo dessa primeira aula plugada foi trabalhar a mesma ideia dos labirintos feitos em sala de aula no papel. Para tanto, utilizamos o jogo "labirinto" do site Blockly Games[13]. Esse jogo possui uma sequência de 10 fases nas quais é necessário guiar o boneco até o alvo marcado. Para solucionar cada um dos desafios e ir avançando no jogo, basicamente é necessário empilhar os códigos em uma determinada sequência que faça com que o boneco se mova pelo percurso previamente oferecido até que alcance o alvo. À medida que as fases vão avançando, novos comandos vão sendo necessários (repetições, tomada de decisão) e vai se tornando cada vez mais desafiador, visto que, além da complexidade do labirinto crescer, acontece uma limitação da quantidade de comandos disponíveis, sendo necessário utilizar blocos de repetição.

A Figura 22 ilustra a quarta fase do jogo labirinto. Destacamo-la devido a sua proximidade com o segundo momento desenvolvido em sala na aula anterior, na qual os alunos utilizaram o comando de giro na atividade do labirinto de forma desplugada.

Logo após a imagem, transcrevemos um diálogo entre os alunos de uma dupla e o pesquisador. Eles estavam trabalhando nessa fase do jogo e, após discussões e tentativas, conseguiram avançar para próxima fase.

Figura 22 - Fase 4 do jogo



Fonte: https://blockly.games

[13] Essa é uma iniciativa do Google objetivando democratizar a programação de computadores. Por meio de pequenos desafios/jogos os alunos vão conhecendo um pouco da linguagem de programação visual. Disponível em: https://blockly.games/?lang=pt-br.

Aluno 1: ele tem que repetir o virar à direita e repetir o virar à esquerda

Aluno 2: não filho, pelo amor de Deus, óh, avança, óh.

Aluno 1: ele tem que repetir...

Aluno 2: Aí tá vendo, é o avança primeiro, óh

Aluno 1: tá, avança, repete, aí vire à esquerda

Aluno 2: não, porque senão ele vai repetir o avança

Aluno 1: não, avança, vira à... vira à esquerda

Aluno 2: à esquerda... pera aí

Aluno 1: avança, de novo

Aluno 2: falta um bloco

Aluno 1: não, mas ele vai avançar, avançar

Aluno 2: é... esse avançar tem... óh, avança, repete. Calma

Aluno 1: todos tem que ter um repetir até faça, até tãnãnã faça

Aluno 2: agora óh, vamo colocar esse daqui pra gente fazer um rascunho. Agora vamo pensar, porque ele tem que avançar, pra ele avançar, ele avança, vira à esquerda

Aluno 1: Não, vira à esquerda não

Aluno 2: Ah é, repete

Aluno 1: Ahhh, tive uma ideia

Pesquisador: Só que vocês têm que tomar cuidado que tem ordem o comando, hem?! É de cima pra baixo que ele vai lendo. [8 segundos de silêncio]. Vai lá, executa e vê o que que vai dar. [4 segundos de silêncio]. O que que rolou? Conseguiu identificar?

Aluno 2: Pera, pera, pera

Aluno 1: Faltou um avança aqui

Pesquisador: Ah, aí, o brother já matou a xarada aí, óh. Acho que não precisa desmanchar hem, é só complementar. [6 segundos de silêncio] Aê... ops [12 segundos de silêncio]. Aê, show de bola

Os alunos foram debatendo o desafio e fazendo diversas tentativas. O pesquisador apenas interviu relembrando-os que os códigos são lidos de cima para baixo e, portanto, era importante pensar na sequência deles. Com isso eles concluíram a fase e seguiram o jogo.

A aula foi bastante dinâmica e os alunos se mostraram muito empolgados devido ao caráter lúdico. Muitos conseguiram avançar nas fases dos jogos apresentando mais familiaridade com a programação visual. Algumas duplas relataram que já haviam visto aqueles jogos tempos atrás (época do projeto de programação para as crianças que ocorreu há 3 anos). Alguns

alunos relataram espontaneamente que observaram semelhanças com as atividades feitas em sala anteriormente.

Ainda nessa aula uma das alunas disse que era mais fácil programar o computador na sala (se referindo a sua dupla na aula desplugada). Ela relatou que mostrava à colega para qual lado deveria caminhar e no computador não era possível essa comunicação, apenas fazer a programação.

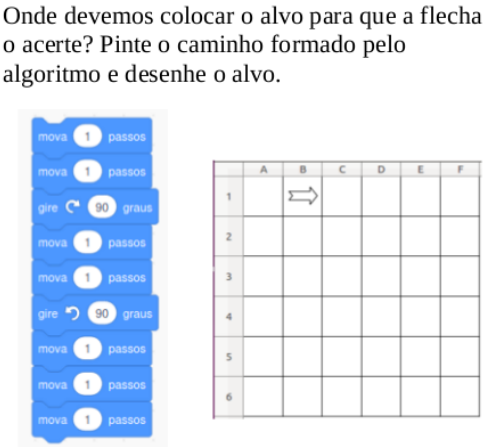
Ao final da aula a maior parte dos alunos havia alcançado o nível 7 ou 8 e se mostraram "decepcionados" com o final do tempo de aula.

4.1.1.4 Aula 4 (28/09/2022) – Atividade desplugada (10:50 às 11:40)

Iniciamos a aula solicitando que os alunos se organizassem em duplas, procurando manter a mesma divisão das aulas anteriores. Não foi possível ter exatamente a mesma formação visto que alguns alunos faltaram da aula anterior e outros faltaram na aula desse dia, assim, aconteceram alguns ajustes.

A primeira atividade dessa sequência consistia em construir o caminho percorrido por uma flecha no quadriculado e indicar onde deveria ficar o alvo para que a flecha o atingisse. A sequência de códigos foi dada e os alunos deveriam ser capazes de interpretar esse código para resolver o problema.

Figura 23 - Enunciado atividade 1

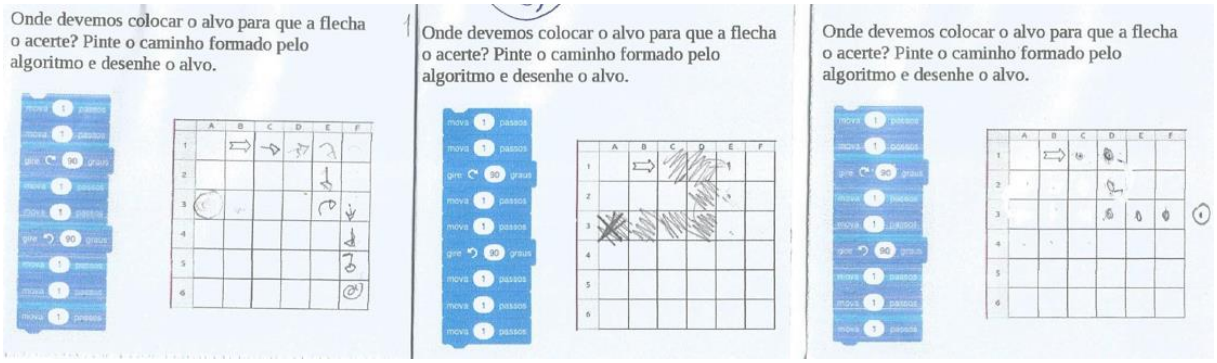


Fonte: o autor

Os alunos rapidamente foram resolvendo o desafio, dizendo ser muito simples, mas logo diziam que não havia o alvo, denotando que não houve uma leitura do enunciado. Após explicação, resolveram, porém, na maior parte dos casos não conseguiram notar a diferença entre os giros que aparece (o primeiro deles o giro é para a direita enquanto o segundo para a esquerda).

A Figura 24 apresenta a resolução das duplas 11, 5 e 2 respectivamente.

Figura 24 - Resoluções atividade 1

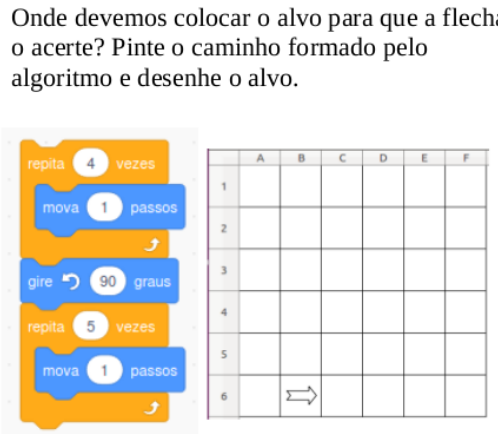


Fonte: o autor

A dupla 11 parece ter interpretado o comando de giro como se ele desse “mais um passo” ao invés de apenas girar, sem sair da célula. Já a dupla 5 não se atentou ao fato de que o segundo comando de giro era diferente do primeiro, ou seja, a flecha iria girar para o lado oposto do primeiro giro feito. Por fim, a dupla de número 2 interpretou corretamente o código.

A segunda atividade, com o mesmo objetivo da primeira (desenhar o alvo e traçar o caminho percorrido pela flecha), utilizava blocos de repetição ao invés de ter o mesmo comando repetidas vezes.

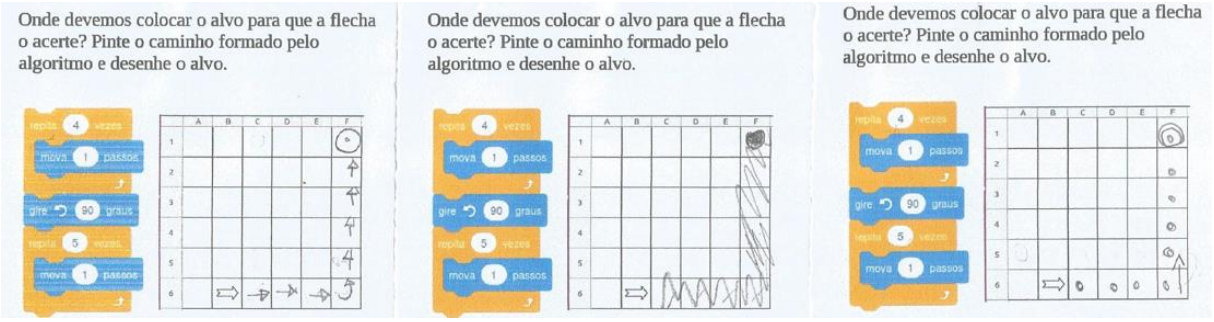
Figura 25 - Enunciado atividade 2



Fonte: o autor

Apesar das estruturas de repetição, os alunos apresentaram mais facilidade para resolver esse desafio, tendo todas as duplas traçado o percurso da flecha corretamente. Na Figura *26* apresentamos a resolução das duplas 11, 5 e 2 respectivamente.

Figura 26 - Resoluções atividade 2



Fonte: o autor

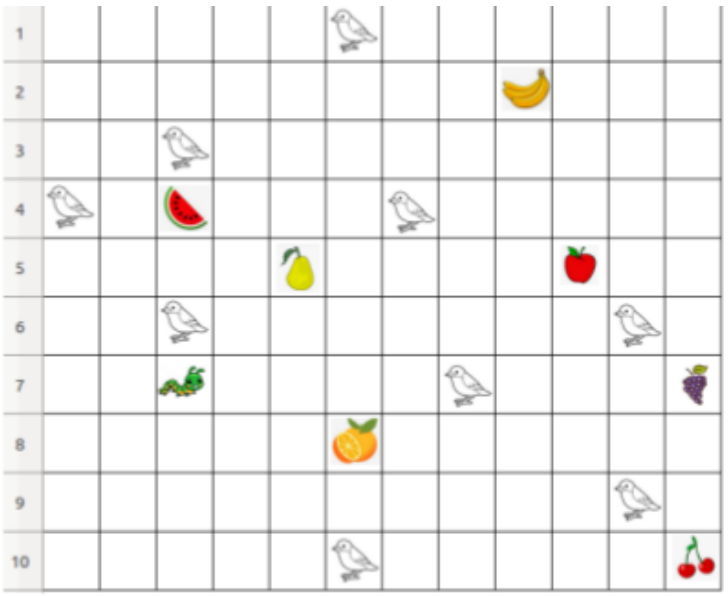
Talvez o fato de haver apenas um comando de giro possa ter facilitado na resolução, não tendo sido um problema a utilização dos blocos de repetição.

A última atividade proposta para essa aula foi o labirinto da lagarta comilona[14]. Essa atividade consistia basicamente em guiar a lagarta até determinadas frutas, segundo cada enunciado, evitando passar nas células que continham pássaros predadores.

As duplas deveriam escrever os caminhos utilizando setas, indicando o percurso da lagarta. Cada dupla de alunos recebeu uma folha conforme ilustrado a seguir.

[14] O nome e a personagem lagarta dessa atividade foram inspirados no livro infantil *Uma lagarta muito comilona* de Eric Carle que minha filha Maria Alice gosta muito.

Figura 27 - Enunciado atividade 3



1) Escreva uma sequência de passos utilizando setas para ajudar a lagarta a:

a) comer a maçã

b) comer a melancia e em seguida a pera

c) comer a banana e em seguida a uva

d) comer as cerejas e em seguida a laranja

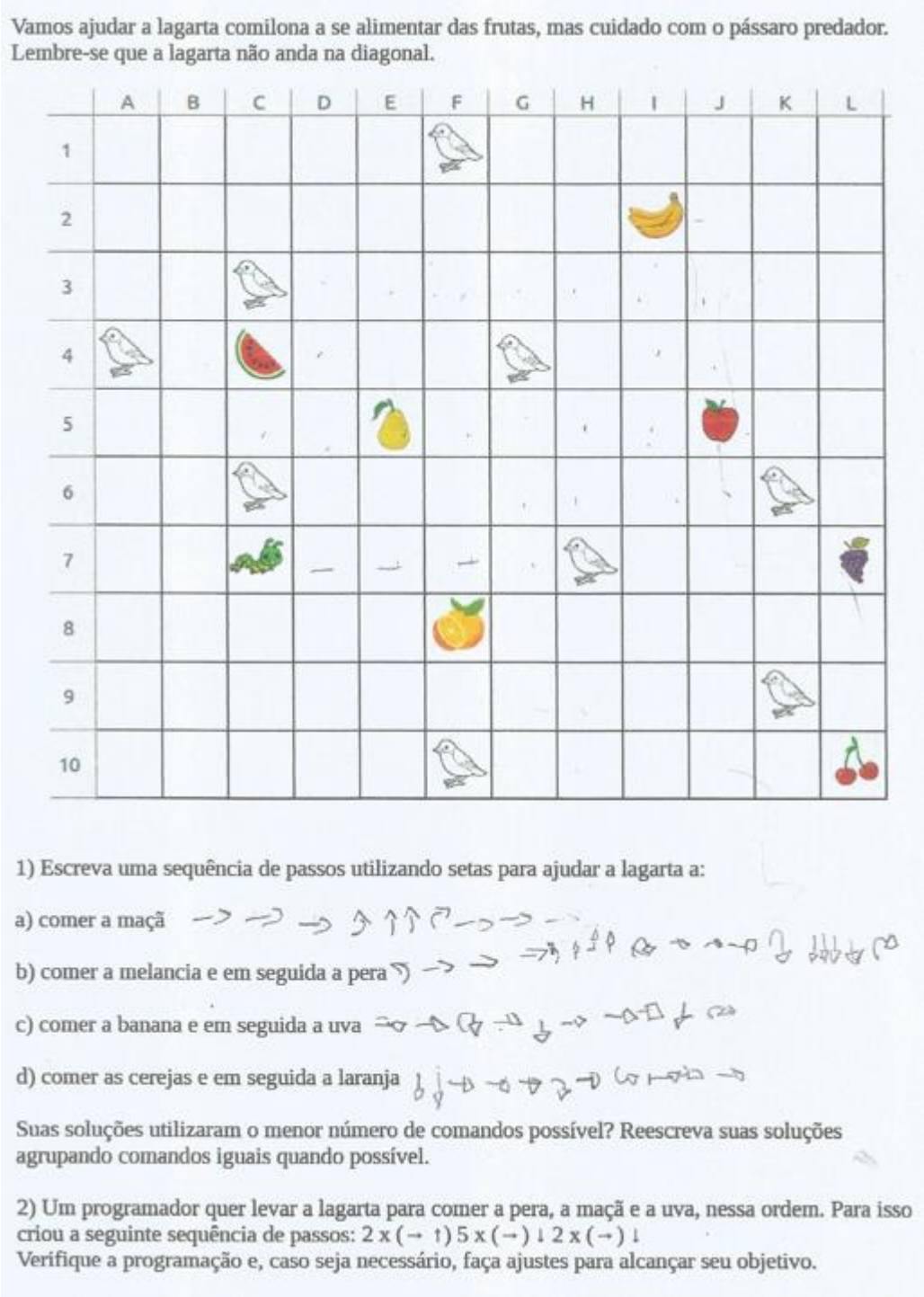
Suas soluções utilizaram o menor número de comandos possível? Reescreva suas soluções agrupando comandos iguais quando possível.

2) Um programador quer levar a lagarta para comer a pera, a maçã e a uva, nessa ordem. Para isso criou a seguinte sequência de passos: 2 x (→ ↑) 5 x (→) ↓ 2 x (→) ↓
Verifique a programação e, caso seja necessário, faça ajustes para alcançar seu objetivo.

Fonte: o autor

Os alunos inicialmente se mostraram empolgados com a atividade, porém, rapidamente cansaram e começaram a dispersar, visto que era uma atividade longa, com muitas perguntas de uma só vez, diferente das desenvolvidas anteriormente. Depois de algum tempo notamos vários alunos dispersos e conversando. Em muitos casos apenas um aluno da dupla resolvia os desafios sozinho. Devido a esse fato, poucos conseguiram concluir a atividade toda, conforme ilustramos com a resolução da dupla 11 na Figura *28*.

Figura 28 - Resolução atividade 3



Fonte: o autor

Observando a resolução dessa dupla, é possível notar que resolveram apenas parcialmente a primeira questão, e sequer conseguiram chegar na segunda questão. Isso aconteceu em outras 4 duplas.

4.1.1.5 Aula 5 (29/09/2022) – Atividade plugada (8:55 às 9:40)

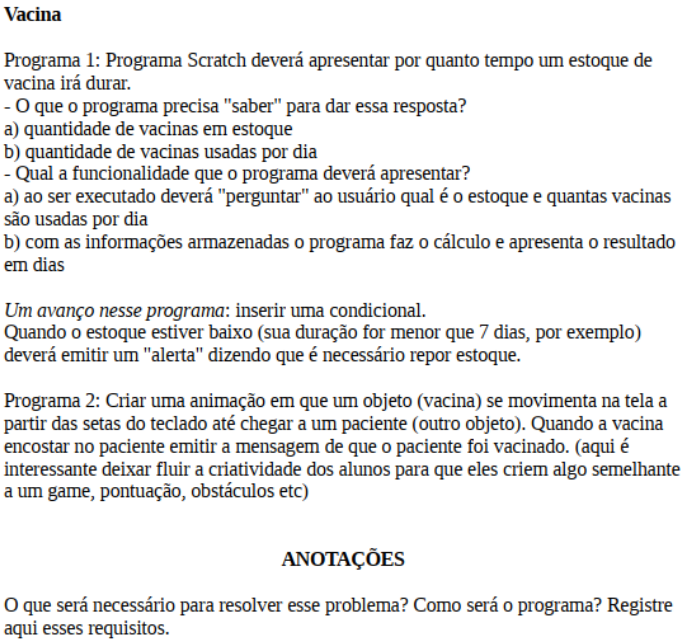
Essa foi a primeira aula das três aulas em que os alunos iriam trabalhar em um projeto Scratch. Inicialmente eles receberam um enunciado de um problema maior que deveria ser resolvido utilizando o Scratch. Devido a conjuntura mundial no momento da coleta de dados, optamos por utilizar como temática geral para o trabalho a COVID-19.

Antes de efetivamente receberem o problema a ser resolvido, foi feita uma abordagem geral sobre a temática, retomando os desafios enfrentados ao longo da pandemia, especialmente no contexto da saúde. Assim, foi dito aos alunos que eles deveriam contribuir com o setor de saúde resolvendo problemas enfrentados por esse setor com o início da pandemia.

Para esse trabalho foi solicitado aos alunos que se dividissem em 5 grupos e, assim, cada grupo receberia um problema diferente a ser resolvido utilizando o Scratch. As temáticas dos problemas foram: vacinas; gasto financeiro com a compra de máscaras para profissionais de saúde; utilização de álcool pelos hospitais; quantidade de leitos disponíveis e quantidade de infectados que represente o pico da pandemia. A solução do problema consistia na construção de um programa Scratch interativo.

A Figura *29* ilustra o enunciado do problema com a temática da vacina.

Figura 29 - Enunciado projeto Scratch



Vacina

Programa 1: Programa Scratch deverá apresentar por quanto tempo um estoque de vacina irá durar.
- O que o programa precisa "saber" para dar essa resposta?
a) quantidade de vacinas em estoque
b) quantidade de vacinas usadas por dia
- Qual a funcionalidade que o programa deverá apresentar?
a) ao ser executado deverá "perguntar" ao usuário qual é o estoque e quantas vacinas são usadas por dia
b) com as informações armazenadas o programa faz o cálculo e apresenta o resultado em dias

Um avanço nesse programa: inserir uma condicional.
Quando o estoque estiver baixo (sua duração for menor que 7 dias, por exemplo) deverá emitir um "alerta" dizendo que é necessário repor estoque.

Programa 2: Criar uma animação em que um objeto (vacina) se movimenta na tela a partir das setas do teclado até chegar a um paciente (outro objeto). Quando a vacina encostar no paciente emitir a mensagem de que o paciente foi vacinado. (aqui é interessante deixar fluir a criatividade dos alunos para que eles criem algo semelhante a um game, pontuação, obstáculos etc)

ANOTAÇÕES

O que será necessário para resolver esse problema? Como será o programa? Registre aqui esses requisitos.

Fonte: o autor

A programação básica do primeiro programa de todos os grupos era muito parecida: criar 2 variáveis para receber os valores e executar uma operação matemática com esses valores (divisão ou multiplicação). Depois disso, seria necessário utilizar um comando de tomada de decisão.

Quando essa atividade foi planejada, era esperado que os alunos anotassem o que deveriam fazer para a construção desse programa e fossem discutindo entre si e tirando as eventuais dúvidas com os professores, porém, nada disso aconteceu. Os alunos apenas leram a "primeira frase" da folha e logo foram fazendo desenhos ou animações sobre (procurando imagem de hospital para ser o plano de fundo, desenhando uma vacina, outro foi fazendo um jornal sobre COVID etc). Nenhuma equipe se preocupou em resolver o problema proposto, mas sim criar algo relacionado à temática.

A transcrição a seguir apresenta o pesquisador procurando explicar a uma das equipes como deveria ser desenvolvido o programa. O programa dessa equipe consistia em dizer após quantos dias um hospital estaria cheio a partir dos dados inseridos pelo usuário (número de leitos disponíveis e quantidade de pessoas que são internadas por dia).

Pesquisador: 20 leitos, vamos supor que ele tenha 20 leitos, e todo dia interna 2 pessoas... pessoal, eu tô falando com vocês também, ouçam aqui. Tem 20 leitos e todo dia interna 2 pessoas. Daqui quantos dias que vai ficar tudo cheio? [4 segundos de silêncio]

Aluno 1: 10

Aluno 2: 10

[outros alunos falam algo inaudível]

Pesquisador: o hospital tem 20 leitos e todo dia interna 2 pessoas, então daqui a 10 dias vai ficar cheio. Isso é o que o programa de vocês tem que fazer. Quando eu mandar executar o programa aqui óh, por exemplo, é... aqui em eventos, quanto eu clicar na bandeirinha verde, o programa vai perguntar assim: quantos leitos tem no hospital? Aí depois você vai digitar a resposta. Depois, quantas, é... quantas pessoas são internadas por dia? Aí você vai digitar a resposta, duas. Aí o programa te fala, então vai gastar 10 dias. Quando executar o programa de novo, eu poderia colocar, eu tenho 30 leitos e interna 2 por dia, aí ele dá um outro resultado. O programa que tem que fazer essa conta, e essa conta vai ser feita usando aqui, óh, operadores, óh, soma, subtração, multiplicação, divisão. Entenderam agora a ideia do que que é pra ser feito?

Os alunos estavam utilizando o Scratch *online*, programando diretamente no site. No meio da aula, devido a problemas em um roteador, a internet parou de funcionar e todos os trabalhos iniciados foram perdidos. A turma logo se dispersou. Alguns começaram a jogar alguns joguinhos, outros a fazerem desenhos e conversarem. A turma logo se dispersou e um "plano B" precisou ser improvisado visto que, ainda nesse dia a mesma turma teria uma segunda aula. Como havia o intervalo (recreio) de 20 min na sequência, esse tempo foi utilizado para improvisar a utilização do Scratch *offline*, instalado nos computadores. Nesse curto espaço de tempo foram ajustados os computadores para que a próxima aula pudesse ser mais produtiva.

4.1.1.6 Aula 6 (29/09/2022) – Atividade plugada (10:00 às 10:50)

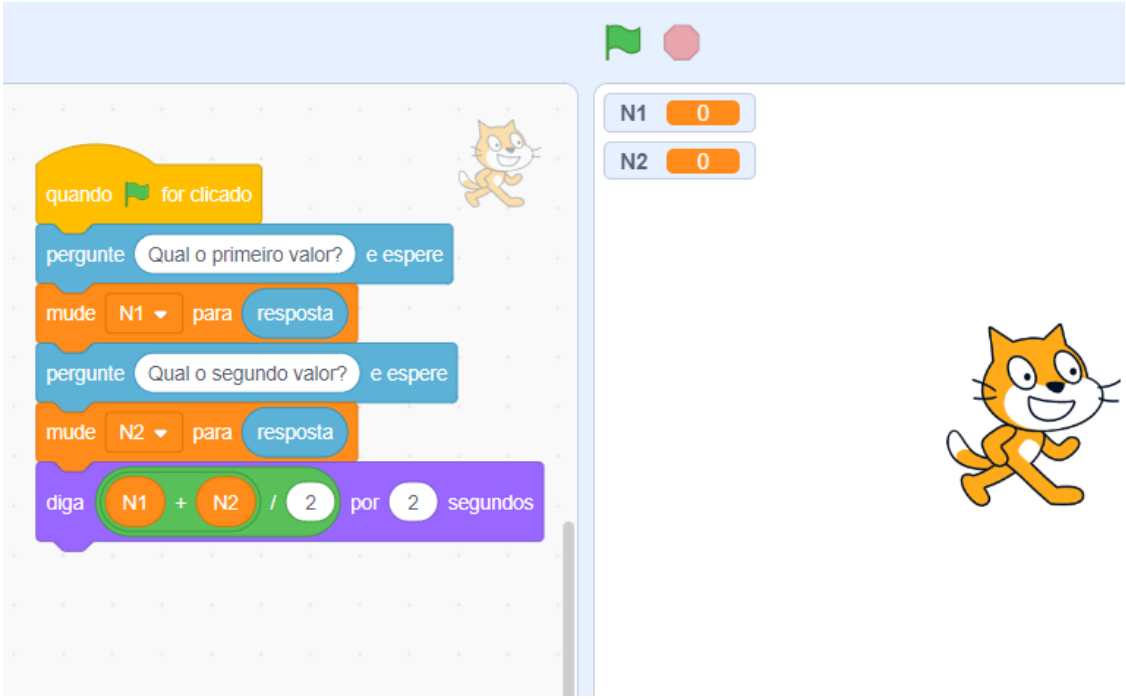
A aula anterior, a primeira que os alunos tiveram contato com o Scratch, permitiu observar que os alunos estavam muito dispersos em relação ao projeto. A intenção de iniciar diretamente com esse projeto o contato com o Scratch era que eles fossem buscando no próprio *software* a solução para o problema e, à medida em que dificuldades surgissem, professores iam auxiliando para que pudessem avançar no trabalho, porém, o que aconteceu foi muito diferente disso. Os alunos não tiveram a persistência esperada para tentar buscar a solução do problema e logo dispersaram por não terem nenhum conhecimento sobre o Scratch. Diante desse fato, na segunda aula, modificamos o planejamento. Iniciamos a aula resolvendo um pequeno probleminha junto com eles para que tivessem o primeiro contato com o Scratch mediados pelo pesquisador, para só então retomarem a resolução do problema COVID-19.

Em uma TV de 42' existente na sala de informática, conectamos um computador e fomos desenvolvendo um pequeno projeto Scratch para calcular a média aritmética de dois números que seriam inseridos pelo usuário. Quando o programa fosse executado ele deveria "perguntar" ao usuário os números e então efetuar o cálculo da média desses números.

Inicialmente fizemos uma discussão com a turma relembrando o que é a média aritmética e como calculá-la. Após esse momento foi apresentado aos alunos o conceito de variáveis no contexto da programação e criamos então duas para armazenar cada um dos valores que o usuário digitasse. Além disso, apresentamos os comandos Scratch para tornar o programa interativo (comando pergunte) e como manipular a resposta inserida pelo usuário.

Após esse contato inicial com tais funcionalidades do Scratch, apresentamos os operadores matemáticos e como fazer a soma e divisão de valores, concluindo assim o programa para o cálculo da média de 2 valores inseridos pelo usuário conforme ilustrado pela Figura *30*.

Figura 30 - Programa Scratch para calcular média de dois valores



Fonte: o autor

Somente depois de retiradas as dúvidas e da conclusão desse programa foi que retomamos o projeto proposto com a temática da COVID-19. Quando retomamos, fizemos uma relação entre esse projeto e o programa que havíamos concluído sobre o cálculo da média, mostrando que o formato de programação era semelhante, visto que necessitaríamos também de duas variáveis e de utilizar os blocos matemáticos para executar a operação necessária. Assim os alunos trabalharam mais direcionados já com uma pequena instrumentalização do Scratch.

Esse "ajuste de rota" no meio do projeto foi fundamental para que pudessem avançar no desenvolvimento do trabalho. Conforme dissemos, era esperado que quando os alunos recebessem o projeto sobre a COVID a ser desenvolvido, fizessem a leitura em conjunto e tentassem buscar a solução por meio de discussões com sua equipe, porém nada disso aconteceu. Após terem sido iniciados no Scratch por meio do programa de calcular a média de dois valores, foi possível notar o avanço que tiveram e então começaram a buscar outros comandos de forma mais independente.

Tal fato nos levou a antecipar esse "ajuste de rota" para a outra turma (7C) que ainda não havia iniciado o projeto. Assim, a primeira aula de Scratch do 7C já foi o problema da média de 2 números e, somente depois de sua conclusão, os projetos sobre a COVID foram entregues. Esses detalhes serão pormenorizados ao relatar as atividades da referida turma.

4.1.1.7 Aula 7 (30/09/2022) - Atividade plugada (10:00 às 10:50)

Essa foi a última aula dos alunos para concluírem seus projetos Scratch. O início da aula foi um pouco confuso visto que os computadores possuem um *software* que o restaura todas as vezes que é reiniciado, assim, os projetos dos alunos tiveram que ser salvos em *pen drive* ao final da aula anterior. Até que pudéssemos distribuir novamente esses trabalhos para os alunos darem sequência levou-se um tempo.

Eles seguiram avançando em seus projetos, porém, era esperado que discutissem com o grupo sobre o trabalho, mas como as equipes tinham cerca de 6 integrantes, a maior parte dos alunos preferiu se dividir em mais computadores. Ficou claro a "decepção" dos alunos ao terem que fazer apenas 1 projeto para uma equipe de 6 integrantes, visto que dessa forma somente um ou dois alunos que iriam manipular os computadores. Assim foi permitido que se dividissem e fizessem o trabalho utilizando mais computadores. Essa foi outra questão que refletimos e ajustamos para a outra aplicação visto que um grupo com essa quantidade de alunos não funcionou bem para o formato de atividade proposto, pois todos queriam manipular o computador.

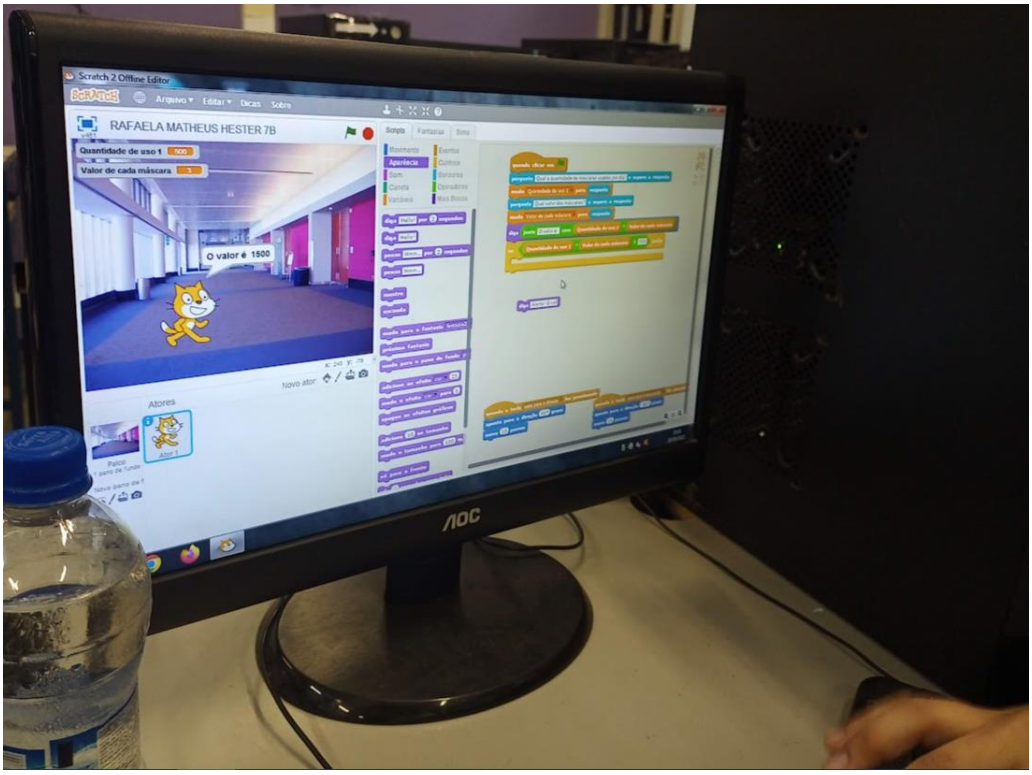
A Figura *31* e a Figura *32* apresentam alguns momentos dos alunos trabalhando em sala na finalização dos trabalhos.

Figura 31 - Turma 7B fazendo projeto Scratch - 1



Fonte: o autor

Figura 32 - Turma 7B fazendo projeto Scratch - 2



Fonte: o autor

Ao final dessa aula os projetos desenvolvidos pelas equipes foram novamente salvos no *pen drive* para que pudessem fazer a apresentação para toda turma na aula seguinte conforme será detalhado na sequência.

4.1.1.8 Aula 8 (05/10/2022) – Apresentação final (10:50 às 11:40)

Nessa aula os alunos tiveram a oportunidade de, na frente da turma, abrir seus projetos, executá-los e comentar um pouco sobre o que fizeram. A princípio, conforme já comentamos, era esperado que fossem apresentados 5 projetos (foi planejado desenvolver o trabalho em 5 equipes de 6 alunos), porém, os alunos se subdividiram e acabaram desenvolvendo trabalhos individuais, em duplas e, assim, foram apresentados 9 trabalhos. É importante frisar ainda que alguns alunos não apresentaram trabalho, ou seja, suas equipes não tiveram nada concluído para mostrar a seus colegas.

A imagem seguinte ilustra o momento em que uma das equipes apresentava seu projeto para a turma.

Figura 33 - Apresentação dos projetos turma 7B



Fonte: o autor

Dos 9 trabalhos apresentados, 4 concluíram a primeira parte solicitada (receber os valores digitados pelo usuário, armazená-los em variáveis e executar o cálculo). Dos outros 5

trabalhos restantes, 3 fizeram uma animação com dois personagens dialogando, 1 fez apenas dois personagens se movimentarem e 1 fez um personagem falando.

Mesmo os alunos que não resolveram o problema proposto (aqueles que fizeram apenas as animações) utilizaram a COVID como temática de seus trabalhos e alegaram que não houve tempo suficiente para concluir seus projetos.

Essa foi a turma que teve a primeira aula incompleta devido a um problema com a internet e, por isso, tiveram apenas duas aulas em contato com o Scratch. Essa questão certamente contribuiu para que não conseguissem avançar muito na proposta.

Na figura a seguir ilustramos o código de uma das equipes que concluiu a primeira etapa solicitada no problema.

Figura 34 - Solução projeto equipe 1



Fonte: o autor

Essa equipe conseguiu criar as duas variáveis necessárias para armazenar os valores inseridos pelo usuário do programa (quantidade de uso 1 e valor de cada máscara) e utilizar o operador de multiplicação para calcular o valor total gasto. Eles ainda utilizaram uma verificação (se) para dar um aviso caso o valor excedesse a quantia de R$500 (valor suposto pelo grupo).

Já a equipe que ilustrada na Figura 35, não se atentou à matemática envolvida para resolver o problema.

Figura 35 - Solução projeto equipe 2



Fonte: o autor

É possível observar que eles calcularam a média entre os valores inseridos (leitos e pacientes). Certamente efetuaram essa operação matemática visto que, conforme já relatamos, antes de eles trabalharem nesse projeto foram introduzidos ao Scratch por meio de um programa que calculava a média entre dois valores inseridos pelo usuário. Dessa forma eles simplesmente criaram variáveis com os nomes condizentes ao problema proposto, mas fizeram a mesma operação de cálculo da média.

A segunda etapa do trabalho consistia em algo mais livre para que os alunos pudessem criar alguma animação ou jogo a partir da temática. Essa etapa nenhuma equipe conseguiu fazer devido ao pouco tempo disponível, conforme já relatamos.

Terminada a apresentação dos trabalhos pelas equipes, passamos a conversar com a turma, procurando fazer uma entrevista coletiva. Alguns trechos dessa entrevista estão apresentados a seguir.

Inicialmente o pesquisador questiona aos alunos sobre as primeiras atividades desplugadas, nas quais um aluno era o programador e o outro o computador.

Pesquisador: Lá naquela primeira aula que a gente fez, na verdade foi no segundo dia que a gente se encontrou, uma atividade em dupla, que tinha lá um quadriculado com uma bolinha azul, um rostinho feliz com um caminho, onde um era o programador e o outro o computador. Vocês lembram dessa atividade, o que que foi mais difícil de fazer nessa atividade pra vocês? (SIC)

[vários alunos falam ao mesmo tempo]

Aluna 1: fazer a atividade em si já foi difícil (SIC)

Aluno 1: mais difícil foi ter que representar a curva pra ele virar pra um lado e ele ter que virar junto com uma seta (SIC)

Após diversas falas dos alunos ficou claro que o ponto de maior dificuldade dessa atividade esteve relacionado com a lateralidade (indicar direita e esquerda), especialmente por estarem assentados um de frente para o outro. Além disso, a questão de representação dos movimentos com as setas também foi apontada.

Na sequência o pesquisador pergunta aos alunos se eles conseguiram observar semelhanças entre as atividades desplugadas, em que um era o computador e o outro o programador com as atividades plugadas nos jogos do Blockly Games.

Aluno 1: os comandos, tem que dar comandos pra... pra chegar em algum certo lugar, determinado lugar (SIC)

Outros alunos concordam e complementam a resposta do colega afirmando que em ambas as atividades era necessário conduzir os objetos por meio de comandos. Após diversos comentários dos alunos, o pesquisador aproveita as falas para emendar uma pergunta questionando se as atividades em sala, nas quais era preciso "programar" o colega, foram mais fáceis ou mais difíceis quando comparadas às feitas na sala de informática, nas quais era preciso programar o computador.

Diversos comentários aconteceram, alguns achando com o computador mais fácil e outros não. Um aluno faz uma colocação interessante, conforme segue:

Aluno 2: É... no computador eu achei mais fácil porque, é... conforme o comado que você coloca, se você colocar o comando certo, ele segue a reta que você coloca. Ele não diferencia, por exemplo, a esquerda da direita, ele vai no comando certo, vamos dizer assim... ele é mais... ele é mais exato. Como a gente tá programando o ser humano, quando a gente tava aqui na sala, é... eles às vezes confundiam, é... o comando que a gente passava pra eles. Tipo, igual a [nome da colega] falou, é... tem vez que ela falava que... é pra virar pra direita e tem vez que a gente vai pra esquerda. Aí a gente não é tão exato igual o computador (SIC)

Essa fala do aluno chama atenção especialmente pelo fato de sua percepção de que, ao programar o colega (o ser humano), era mais complexo, pois há o processo de interpretação do

colega em relação ao comando, ou seja, o outro questiona, pensa, se confunde, diferente do computador que simplesmente executa exatamente o que foi mandado, sem nenhum tipo de interpretação.

4.1.2 7C

A turma 7C é composta por 28 alunos com idades próximas a 12 anos. Conforme relatado pela professora de Matemática, no geral essa é uma turma mais centrada quando comparada ao 7B, com exceção de alguns casos isolados.

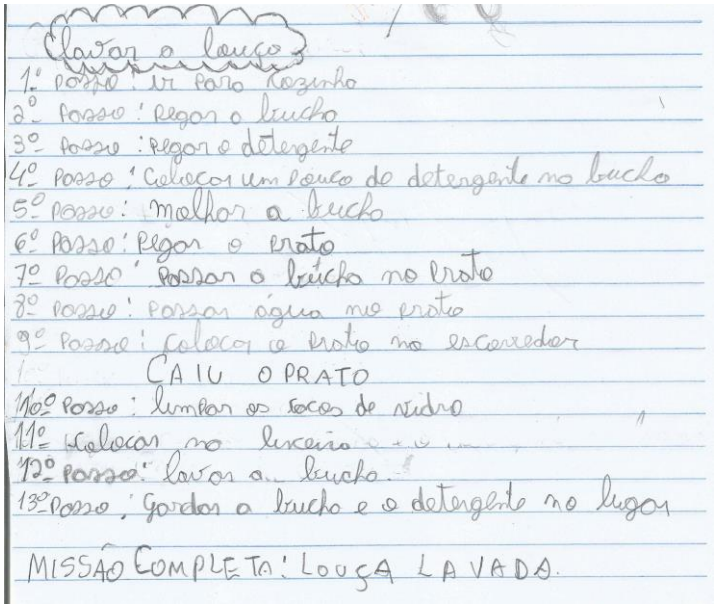
4.1.2.1 Aula 1 (21/09/2022) - Introdução (10:00 às 10:50)

A introdução se deu de forma muito parecida com a da turma 7B, porém, o vídeo do robô aspirador foi apresentado primeiro e, após o debate do vídeo é que foi dado o exemplo da soma de dois números.

O debate sobre o algoritmo do robô na primeira turma (7B) foi mais proveitosa. Os alunos participaram mais e se mostraram mais empolgados com o fato de “tentar descobrir” qual seria a programação que estava por dentro do robô. Ao que parece, ter dado o exemplo do algoritmo da soma de dois números antes do vídeo facilitou o entendimento dessa "sequência de passos" a serem dados e isso acabou despertando mais argumentos sobre como deveria ser o algoritmo do robô.

Após o exemplo de ensinar um robô a "escovar os dentes", foi solicitado que construís-sem uma sequência de passos como se estivessem ensinando a um robô fazer alguma atividade rotineira. A seguir ilustramos a sequência de passos desenvolvida por dois alunos.

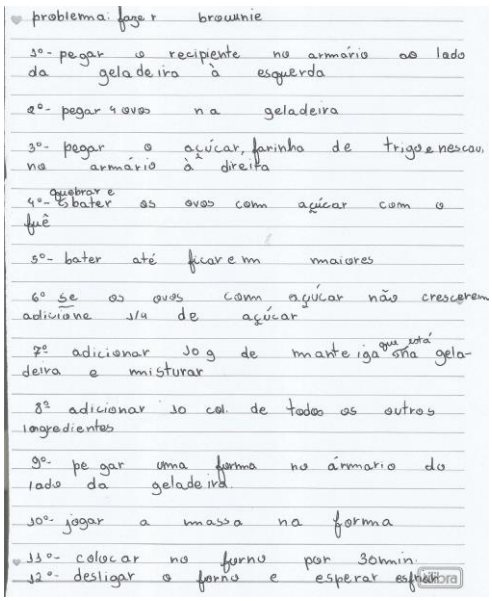
Figura 36 - Atividade 1 aluno A



Fonte: o autor

Esse aluno traz uma sequência de passos de como lavar louça (essa tarefa foi apresentada por diversos alunos) conforme apresentado. Já a segunda aluna, ilustrada na Figura 37, apresenta os passos de como fazer um brownie, indicando de forma detalhada os ingredientes e suas localizações para que o "robô" fosse capaz de encontrá-los.

Figura 37 - Atividade 1 aluno B



Fonte: o autor

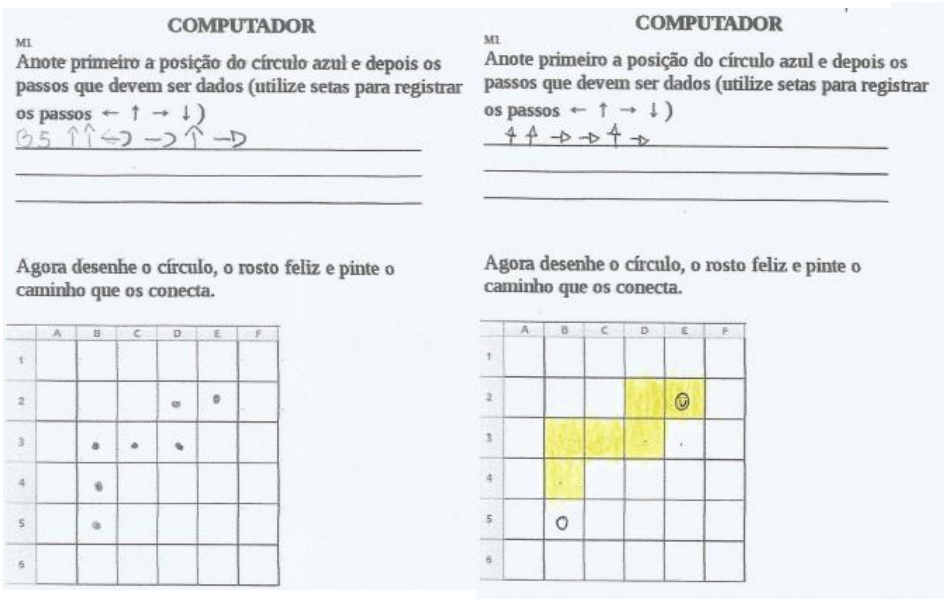
Após alguns comentários com os alunos sobre o trabalho feito por eles encerramos o primeiro encontro.

4.1.2.2 Aula 2 (22/09/2022) – Atividade desplugada (10:00 às 10:50)

As atividades desenvolvidas nessa turma foram as mesmas daquelas desenvolvidas no 7B. Das 13 duplas que participaram dessa atividade, 4 indicaram a posição inicial do círculo e utilizaram as setas para conduzi-lo pelo caminho; outros 4 utilizaram as setas sem indicar a posição inicial e as outras 5 utilizaram outras estratégias, especialmente escrevendo o caminho por extenso.

A seguir trazemos algumas resoluções dessas atividades pelos alunos.

Figura 38 - Resolução atividade 1 - A e B

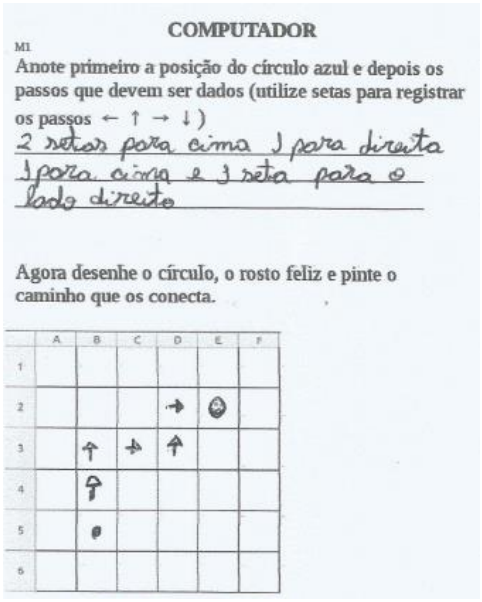


Fonte: o autor

A ilustração à esquerda da Figura 38 mostra a solução dada pela dupla 12, enquanto a ilustração à direita, a solução dada pela dupla 2. A primeira indicou a posição inicial do círculo, enquanto a segunda não. Ambas representam a movimentação do círculo por meio das setas, conforme era esperado.

A estratégia de escrever por extenso foi utilizada por 4 duplas, conforme ilustramos a solução da dupla 5.

Figura 39 - Resolução atividade 1C

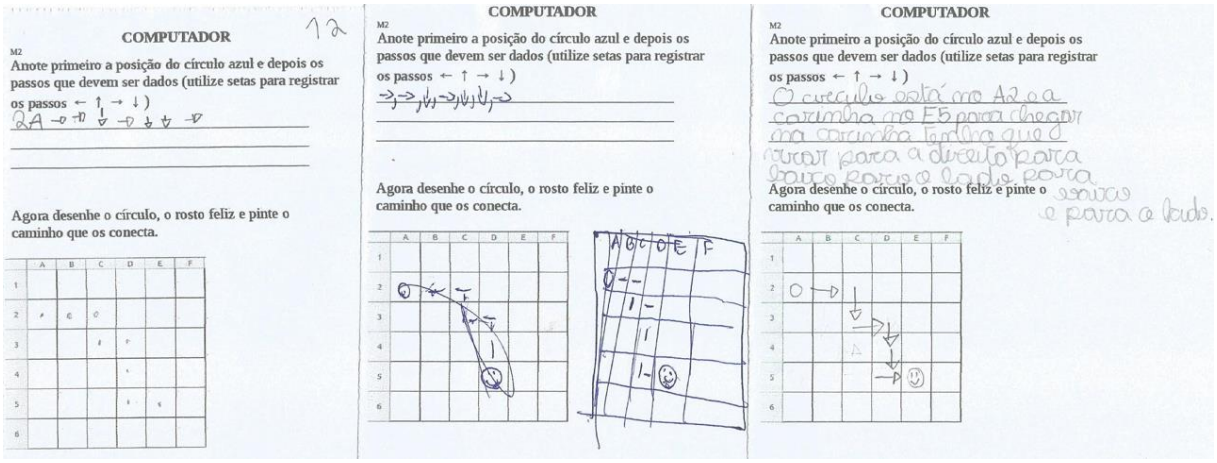


Fonte: o autor

É interessante notar que a resolução das atividades no segundo momento (quando um labirinto diferente é entregue para a dupla e os papéis dos alunos são trocados) se deu de forma muito parecida com o primeiro momento, deixando subentender que houve uma melhor sincronia entre os integrantes das duplas.

Na Figura *40* apresentamos a resolução do momento 2 das duplas 12, 2 e 5 respectivamente.

Figura 40 - Resolução atividade 2 A, B, C



Fonte: o autor

É possível notar a semelhança com a resolução do primeiro momento: a dupla 12 indicou a posição inicial e fez o trajeto por setas; a dupla 2 indicou o trajeto por setas e, por fim, a dupla 5 escreveu por extenso o caminho a ser percorrido.

4.1.2.3 Aula 3 (28/09/2022) – Atividade plugada (10:00 às 10:50)

Assim como na turma 7B os alunos tiveram contato com os jogos Blockly Games, objetivando estudar de maneira plugada os labirintos vistos em sala na aula anterior, bem como trabalharem com comandos mais avançados.

Uma dupla de alunos chama o pesquisador dizendo estar com dificuldades na fase 6 do jogo que está ilustrada na Figura *41*.

Figura 41 - Fase 6 do jogo



Fonte: https://blockly.games

Ao chegar até o computador dos alunos, o pesquisador observa que eles estavam tentando usar os blocos para solucionar exatamente esse labirinto, com várias repetições. Parece que eles não tinham entendido que, na verdade, ao utilizar o bloco de verificação, a solução

poderia servir para outros labirintos visto que o boneco verifica o caminho antes de percorrê-lo. Além do mais, há limitação na quantidade de blocos e, por isso, seria impossível utilizar uma grande quantidade de comandos repetidos, sendo então obrigado a utilizar os comandos de repetição e verificação. A seguir está transcrita a explicação que o pesquisador fez para os alunos.

Pesquisador: Deixa eu explicar uma coisa pra vocês (SIC)

Aluno 1: É porque ele chega aqui... (SIC)

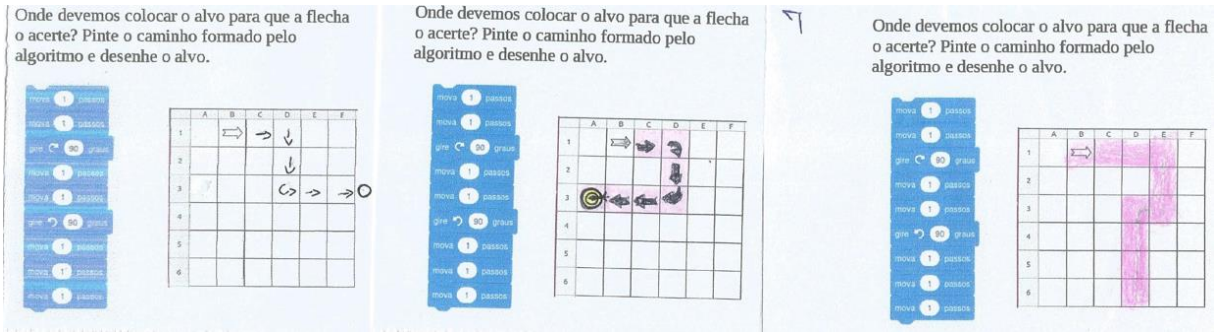
Pesquisador: É... mas olha só o que tá acontecendo. Vocês estão querendo criar uma solução específica para esse mapa. Só que você tem que criar uma solução que seja abstrata, que funciona pra tudo. Então, se você observar o bonequinho tem três alternativas: ou ele avança, ou ele vira pra esquerda, ou ele vira pra direita, certo? Então você vai ter que fazer as verificações. Quando é que ele vai avançar? Quando tiver caminho pra frente. Quando é que ele vira pra esquerda? Quando tem caminho pra esquerda. Quando é que ele vira pra direita? Quando tem caminho pra direita. Entenderam? Então, tá tudo certo o começo: se caminho à esquerda, vire à esquerda, senão, você vai ter que colocar outra verificação aqui dentro, você vai ter que colocar um outro se. Você só vira pra direita, coloca esse se aí pra dentro... isso, óh... se caminho à direita, agora, porque à esquerda você já olhou, vire para a direita, mas é uma vez só, não é duas não... isso... agora aqui embaixo é, se ele não tem caminho nem pra esquerda e nem pra direita, é o que que ele tem que fazer? Avançar! (SIC)

Após essa explicação os alunos pareceram ter entendido e seguiram avançando nas próximas fases de maneira independente.

4.1.2.4 Aula 4 (29/09/2022) – Atividade desplugada (10:50 às 11:40)

Assim como aconteceu com a turma 7B, a maior parte dos alunos na primeira atividade fez confusão com a questão do giro. Das 12 duplas que estiveram presentes nessa aula, apenas 2 conseguiram observar corretamente os comandos de giro; 6 duplas não observaram que o segundo giro era diferente do primeiro e, por fim, as outras 4 duplas fizeram interpretações diferentes dos códigos. Na Figura *42* ilustramos respectivamente: uma dupla que interpretou corretamente todos os códigos (dupla 1); uma dupla que não observou que o segundo giro era diferente do primeiro (dupla 8) e uma dupla que fez uma interpretação diversa (dupla 7).

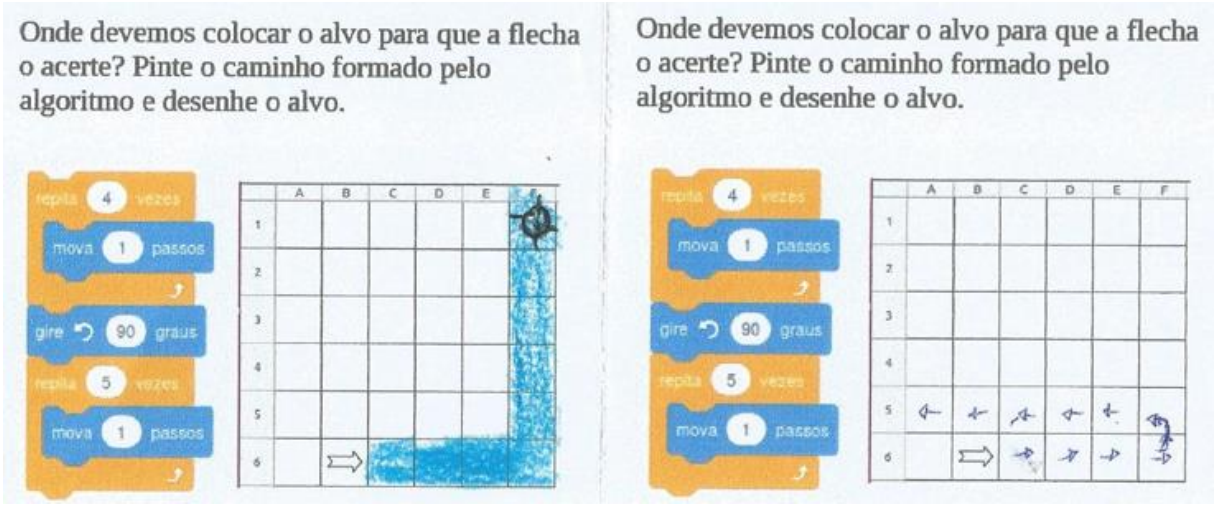
Figura 42 - Resolução atividade 1 aula 4



Fonte: o autor

Já o segundo momento da aplicação, 9 das 12 duplas resolveram corretamente o labirinto. A Figura *43* traz a solução correta dada pela dupla 13 e a solução equivocada dada pela dupla 6.

Figura 43 - Resolução atividade 2 aula 4



Fonte: o autor

É possível inferir que a dupla 6 (e as outras 2 duplas) que cometeram esse erro podem ter se confundido com a setinha que aparece no bloco do comando de repetição (setinha branca no bloco laranja). Talvez esses alunos possam ter interpretado que essa seta estava indicando um outro giro, assim, duas setas sequenciais daria um giro de meia volta, o que daria uma resolução semelhante a apresentada por essas duplas.

No labirinto da lagarta comilona também ocorreram as mesmas dificuldades da primeira turma. Os alunos iniciaram bem a resolução, mas logo se dispersaram. A atividade foi se tornando cansativa e a maior parte da turma não conseguiu conclui-la.

O diálogo transcrito a seguir mostra o pesquisador atendendo a uma dupla que estava resolvendo a primeira questão do labirinto da lagarta comilona que consistia em direcionar a lagarta até a maçã.

Pesquisador: E aí, carinhas, como é que tá aí? (SIC)

Aluno 1: Tentando fazer... (SIC)

Pesquisador: Fácil? Conseguiram entender a dinâmica, como é que é pra fazer? (SIC)

Aluno 1: Mais ou menos... (SIC)

Pesquisador: Mais ou menos? (SIC)

Aluno 2: [inaudível]... dois x (SIC)

Pesquisador: Porque dois x aí, o que que é isso? (SIC)

Aluno 2: Porque a lagarta vai andar duas vezes (SIC)

Pesquisador: Beleza, então você colocou aqui, a lagarta vai comer a maçã, então ela vai andar um, dois, chegou aqui, e agora? (SIC)

Aluno 2: Sobe... (SIC)

Pesquisador: Subiu... Chegou aqui (SIC)

Aluno 1: Aí vai virar à direita... não, uma, duas, três, quatro..., quatro, (SIC)

Aluno 2: Pode subir cinco... (SIC)

Aluno 1: quatro vezes pra... pra frente, aí depois vai fazer o giro, pra ela subir pra cima, a setinha pra cima, e depois vai passar na maçã (SIC)

Pesquisador: Pronto, resolveu! (SIC)

A princípio os alunos relataram estar com um pouco de dificuldade, mas com um pequeno apoio do pesquisador conseguiram resolver a atividade e seguiram avançando.

4.1.2.5 Aula 5 (30/09/2022) – Atividade plugada (8:55 às 9:40)

Devido às dificuldades iniciais observadas na turma anterior (7B), optamos por fazer um pequeno ajuste na primeira aula plugada referente ao projeto Scratch. Ao invés de iniciarmos diretamente trabalhando no projeto, optamos por iniciar os alunos no *software* Scratch por meio de um programa para calcular a média aritmética entre dois números inseridos pelo usuário (assim como fizemos na segunda aula da turma anterior).

Fomos construindo um debate com os alunos, buscando relembrá-los os conceitos básicos da matemática sobre o que seria a média aritmética entre valores e como fazer esse

cálculo. Após esse momento, o pesquisador foi conduzindo a resolução desse problema no Scratch junto aos alunos, resolvendo em um computador que utiliza uma TV como monitor e toda a classe acompanhando, conforme Figura *44*.

Figura 44 - Pesquisador conduzindo projeto Scratch



Fonte: o autor

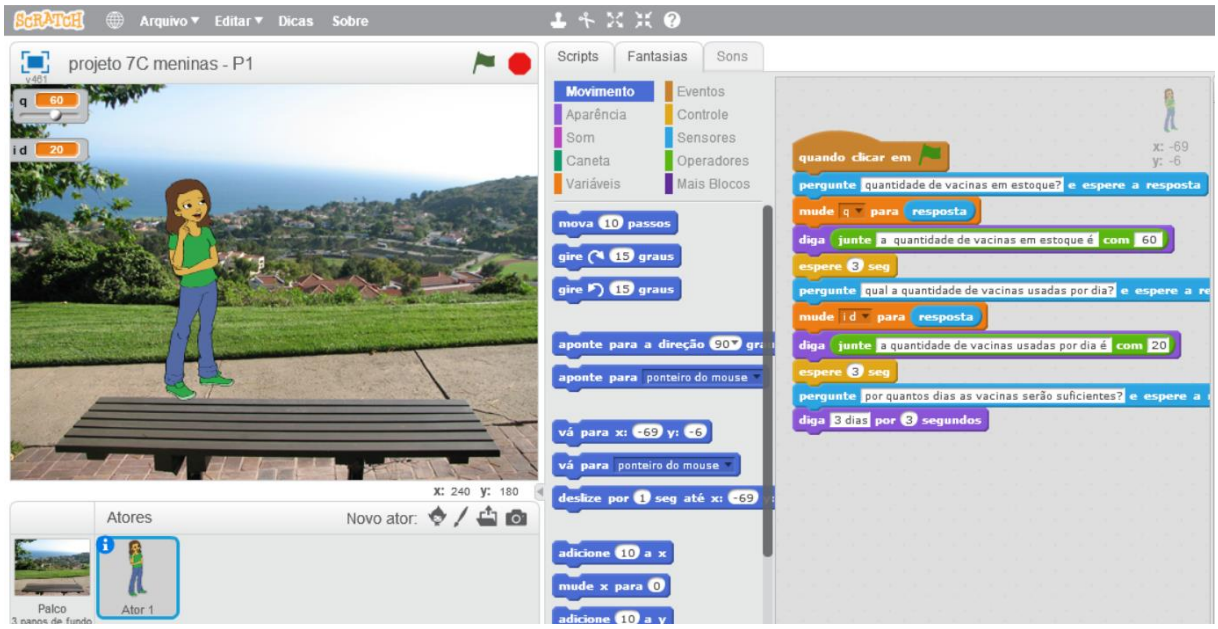
Com esse programa inicial foi possível apresentar aos alunos alguns comandos básicos do Scratch, como operadores matemáticos, utilização de variáveis, blocos de pergunta e resposta e outros que envolvem a fala de personagens. Tendo os alunos esse conhecimento básico desse *software* de programação, na aula seguinte iniciamos o projeto COVID-19 com as equipes conforme relato a seguir.

4.1.2.6 Aula 6 (30/09/2022) – Atividade plugada (10:50 às 11:40)

Na segunda aula de Scratch os alunos iniciaram o projeto com a temática da COVID-19. Ficou muito claro a melhor desenvoltura dessa turma em relação à primeira com o projeto. Certamente o fato de eles terem sido iniciados no Scratch antes de terem tido contato com o trabalho que desenvolveriam facilitou o avanço, visto que já tinham uma noção básica do funcionamento do programa bem como de alguns comandos que poderiam utilizar no Scratch.

Na Figura *45* apresentamos o projeto de uma das equipes ao final da primeira aula Scratch para o desenvolvimento do projeto sobre COVID-19.

Figura 45 - Projeto Scratch aula 6



Fonte: o autor

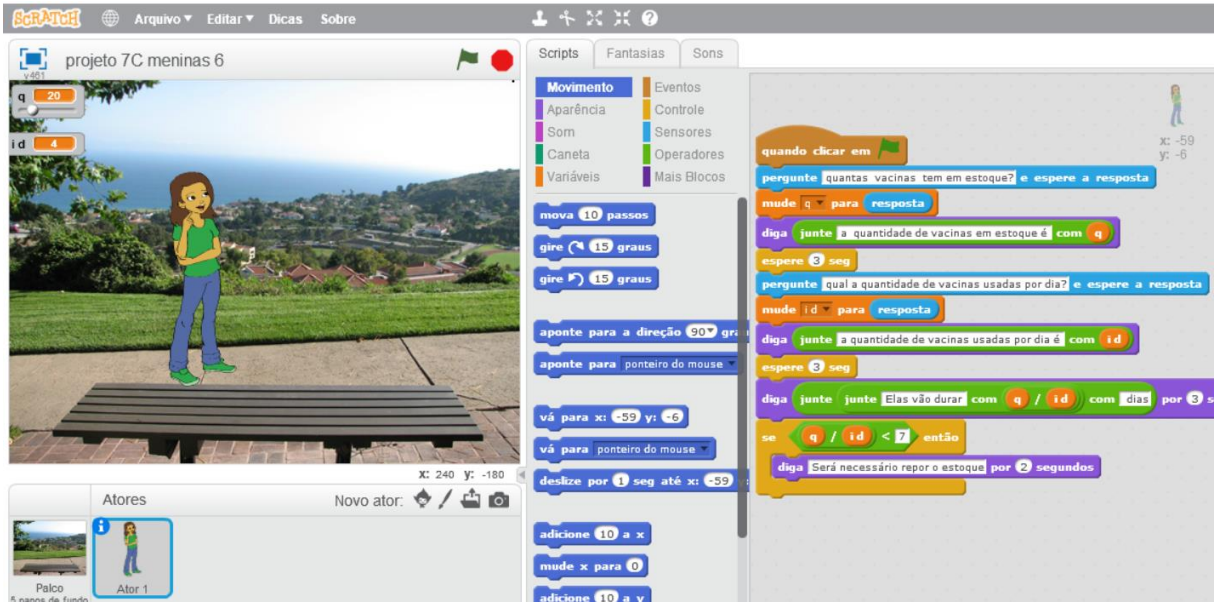
É possível observar que essa equipe conseguiu, em apenas uma aula, utilizar variáveis, comandos de fala e de pergunta e ainda modificar a parte visual do programa (troca de personagem e plano de fundo).

Alguns probleminhas na programação foram ajustados na aula seguinte, conforme será apresentado na sequência.

4.1.2.7 Aula 7 (05/10/2022) – Atividade plugada (8:55 às 9:40)

Essa foi a última aula que a turma teve para proceder à finalização de seus trabalhos. Algumas equipes não conseguiram concluir todo o trabalho, mas outras sim. A Figura *46* apresenta a mesma equipe ilustrada na seção anterior.

Figura 46 - Projeto Scratch aula 7



Fonte: o autor

É possível observar um considerável avanço no trabalho desses alunos em relação à utilização dos operadores matemáticos bem como das variáveis. Além disso, nessa versão a equipe conseguiu utilizar o comando de verificação “se” para a tomada de decisão em relação a emitir um alerta.

Um fato que ficou constatado pelo pesquisador após finalizada a última aula do projeto na sala de informática foi que desenvolver esse tipo de projeto com equipes de 5 ou 6 alunos não foi uma boa escolha. Em sua maioria, havia 2 ou 3 alunos mais envolvidos na resolução do problema enquanto os outros, na maior parte das equipes, estavam dispersos, conversando e, muitas vezes, atrapalhando os colegas que estavam engajados na resolução. Essa constatação ensejou uma mudança nesse ponto para a próxima aplicação da atividade com a turma 7A. Os detalhes dessa aplicação bem como os ajustes sofridos pela sequência de aulas serão detalhados mais à frente.

4.1.2.8 Aula 8 (05/10/2022) – Apresentação final (10:00 às 10:50)

Nessa aula, assim como na outra turma, as equipes apresentaram seus trabalhos aos colegas. Nessa apresentação os alunos tiveram a oportunidade de mostrar o que fizeram e discutir parte dos códigos. Nem todas as equipes se sentiram à vontade para fazer a apresentação, assim, respeitamos a decisão dos alunos em ir até a frente da classe para mostrar seus trabalhos.

A Figura *47* é referente à equipe que ilustramos os códigos anteriormente.

Figura 47 - Apresentação projeto equipe A



Fonte: o autor

Ao apresentar seu trabalho, a equipe iniciou explicando seu problema e como fez a resolução.

Aluna 1: A gente pegou o roteiro das vacinas. A gente primeiro tinha que perguntar quantas vacinas tinha o estoque, e o objetivo era saber, se... quan... quantos dias elas vão durar. Se for menos de 7, seria necessário repor o estoque. Aí a gente vai executar aqui (SIC)

[a aluna vai até o computador, executa o programa e mostra o código para os colegas explicando seu funcionamento]

A equipe que ilustramos na Figura *48* não conseguiu concluir o projeto. Pela apresentação das alunas, fica claro a preocupação delas com a parte visual do programa e de se construir uma espécie de game.

Figura 48 - Apresentação projeto equipe B



Fonte: o autor

Aluna 1: Então... o nosso projeto aqui, ele foi um, a gente diferenciou um pouco porque, era... ele deu um pequeno bug no final e por isso a gente não vai conseguir apresentar muito ele, mas eu vou falar o que era pra ele fazer. Seguinte, tem um médico, não era pra ser esse médico, era um outro [fala com risos] *que ele ia calcular o preço de máscaras por dia* [risos]. *A utilização de quantas máscaras por dia mais o preço que ela custa hoje em dia, né. A gente calculou mais ou menos que seriam 2 máscaras, no máximo, por dia já... seria o suficiente e botamos em média 4 reais, porque a gente foi pesquisar um pouco só que ficou meio balanceado, entre mais caro e mais barato, então a gente deixou 4 reais mesmo. E o objetivo final era onde a gente teria um... uma espécie de um mini game onde a gente teria que conectar máscara a...a... máscara ao médico, que é essa imagem que vocês estão vendo [...] (SIC)*

A aluna termina a apresentação retomando que o projeto está com um problema e por esse motivo não seria possível fazer sua execução.

Finalizadas as apresentações, assim como na turma anterior, procedemos a uma entrevista coletiva.

Inicialmente foi questionado aos alunos sobre as primeiras atividades desplugadas em que eles trabalharam em duplas, sendo um aluno o computador e o outro programador. Um aluno diz que foi “facílimo” fazer a atividade, porém, outro faz uma ponderação.

Aluno 2: fácil, mas... não foi tão fácil porque teve desencontros de... tipo... entender o assunto, entender a pronúncia, do enunciado, do negócio, teve alguns desencontros problemas. Por exemplo, a [nome da colega] *tava falando ‘não sei o que não sei o que lá’* [expressão para dizer qualquer coisa] *e eu tava falando que era aquilo, não... eu só tô dando um exemplo aleatório, é só um exemplo. Aí acaba se desencontrando aí tinha que confirmar [...] (SIC)*

A partir dessa fala do aluno o pesquisador aproveita a oportunidade para questionar aos alunos como esses “desencontros” eram acertados com o computador, visto que em sala, com o colega, eles conversavam para acertar os caminhos a serem seguidos, mas no computador isso não era possível. A partir desse questionamento os alunos rapidamente se manifestaram.

Aluna 1: tentativa e erro (SIC)

Outro aluno complementa a resposta dizendo que pode ir fazendo, fazendo, tentando e acaba concluindo que “é, por tentativa e erro mesmo”.

Na sequência é questionado aos alunos sobre semelhanças entre as atividades desplugadas e as plugadas no Blockly Games. Os alunos afirmam que são “a mesma coisa”. Outro já afirma que possuem o mesmo objetivo, porém, ao invés de utilizarem as setinhas (se refendo a atividade desplugada), diz que tinham que montar por bloquinhos.

Após algumas discussões com os alunos, eles afirmam que a programação de computador é mais difícil do que programar o colega. Conforme relataram, com os colegas você consegue conversar e ir mostrando o que quer que ele faça, já com o computador não, pois ele executa exatamente o que você manda, sem nenhum discernimento.

4.2 Segunda aplicação da sequência didática

Essa segunda aplicação da sequência didática sofreu algumas alterações quando comparada com a primeira. Alguns ajustes se mostraram necessários ao efetuar a primeira aplicação, tanto nas atividades quanto na ordem que foram aplicadas e no formato de distribuição dos alunos durante os trabalhos.

A primeira aula introdutória bem como as outras duas aulas desplugadas (com as atividades em duplas) foram desenvolvidas no mesmo formato das turmas anteriores, com exceção da atividade da lagarta comilona que sofreu alguns ajustes que serão detalhados mais à frente. Já a primeira atividade desplugada em duplas (na qual um aluno era o computador e o outro o programador) iniciamos alterando a posição das duplas. Ao invés de colocarmos um aluno de frente para o outro, solicitamos que ficassem um ao lado do outro, buscando minimizar a dificuldade em relação à lateralidade (virar à direita ou à esquerda), porém, essa alteração não funcionou bem e logo foi desfeita. Em relação às aulas plugadas, a primeira delas não ocorreu mais no Blockly Games, mas sim diretamente no Scratch com construção de labirintos. As demais, referentes ao desenvolvimento do projeto COVID seguiram no mesmo formato, exceto a quantidade de alunos por equipe que foi reduzido. Todos os detalhes serão apresentados na sequência.

4.2.1 7A

Essa turma é composta por 30 alunos com idades próximas a 12 anos. Conforme relatado pela professora de matemática, comparada com as outras, essa é uma turma mais tranquila de se trabalhar.

4.2.1.1 Aula 1 (19/10/2022) - Introdução (10:00 às 10:50)

Durante a discussão inicial sobre o PC com os alunos, foi possível observar que, diferente das demais turmas, eles não tinham nenhuma noção do que poderia ser o PC e nem tinham contato com programação de computadores. Houve discussão com a turma sobre algoritmos como sendo uma sequência de passos ordenados para solucionar determinado problema.

Na sequência foi feita a operação de soma de 2 números com 3 dígitos para que os alunos pudessem "ensinar" a resolver e avançamos no entendimento do que seria um algoritmo. Após esse momento foi apresentado o vídeo do robô aspirador e então discutido um possível algoritmo para que ele pudesse executar essa tarefa.

A finalização da aula se deu com os alunos descrevendo em uma folha de caderno uma tarefa rotineira no formato de algoritmo, como se estivessem programando um robô para executar essa tarefa. Assim como nas turmas anteriores o pesquisador deu como exemplo "como escovar os dentes", apresentando o passo a passo para tal (ir ao banheiro, abrir gaveta do armário, pegar escova e pasta etc).

Esse momento da aula foi curioso pois, mesmo tendo explicado aos alunos que a ideia era pensar em um robô para resolver determinada tarefa cotidiana (um problema), como o exemplo de escovar os dentes, alguns alunos disseram que queriam um robô que resolvesse qualquer tarefa. Eles ainda não tinham compreendido que era necessário ter uma tarefa exata para então dar a esse robô os passos exatos para resolver a questão. Ao caminhar pelas carteiras e observar as atividades dos alunos procurando instiga-los a avançar no trabalho, o pesquisador observou que um aluno havia escrito apenas uma frase na folha e disse ter terminado. Foi então que o pesquisador o interrogou.

Pesquisador: E aí, irmãozinho. O que que você fez aí?(SIC)

Aluno: Um robô companheiro(SIC)

Pesquisador: Hem? (SIC) (pesquisador demonstra espanto sem entender o que o aluno disse)

Aluno: Um robô companheiro(SIC)

Pesquisador: Me explique, vamo lá(SIC)

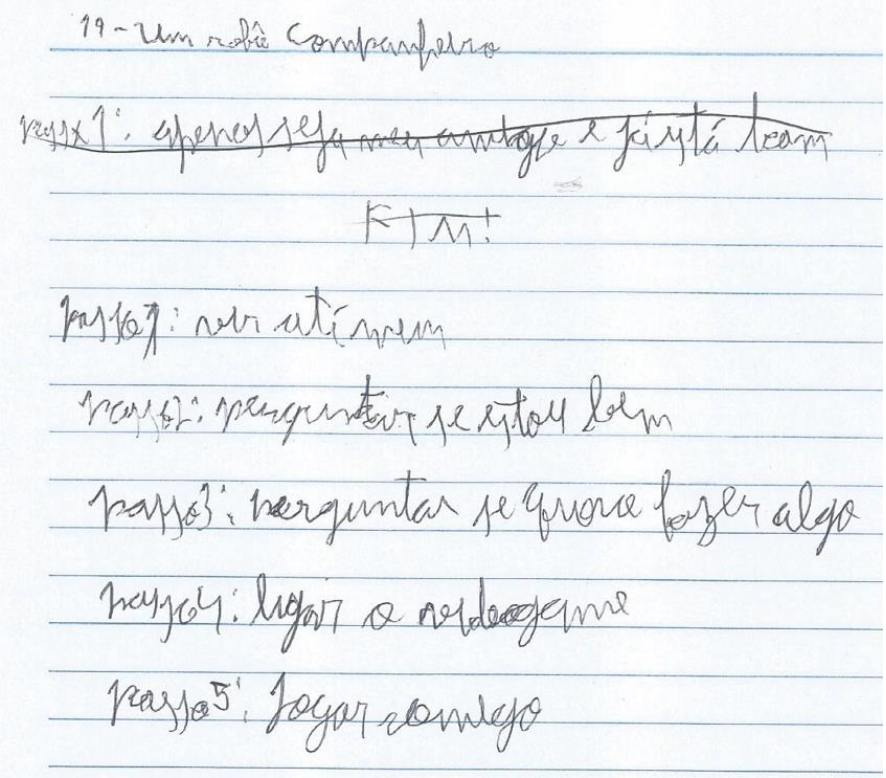
Aluno: É... [inaudível] Porque, se... tiver um robô que converse comigo no dia a dia, já tá bom, não precisa de mais nada(SIC)

Pesquisador: Beleza... Mas aí, o que que tá escrito aí? (SIC)

Aluno: Apenas seja meu amigo e já está bom(SIC)

Conforme explicado no início da atividade, a intenção era que os alunos pensassem em alguma tarefa de seu cotidiano e então "ensinassem" a um robô resolver essa tarefa, descrevendo o passo a passo de como fazê-lo. No intuito de não obrigar o aluno a mudar totalmente aquilo que havia feito, foi solicitado que escrevesse de forma mais completa, em forma de passos o que ele havia imaginado. O trabalho do aluno está apresentado na ilustração seguinte:

Figura 49 - Resolução atividade 1



Fonte: o autor

Outros quatro alunos descreveram passos na mesma “linha” de raciocínio desse aluno, construindo um robô para “ouvir e opinar sobre” e outro até disse ser um robô psicólogo. Os demais alunos da turma criaram algoritmos de tarefas cotidianas conforme o esperado (robô para lavar louça; fazer dever; coletar cocô de cachorro etc).

4.2.1.2 Aula 2 (19/10/2022) – Atividade desplugada (10:50 às 11:40)

Iniciamos a aula solicitando que os alunos ficassem em duplas, com uma carteira ao lado da outra e não mais com uma carteira de frente para a outra. A intenção era que a confusão com a lateralidade dentro da atividade apresentada pelas turmas anteriores fosse mitigada, porém, surgiu um novo problema que foi o excesso de conversa. Como não tínhamos um contato prévio com a turma, não é possível afirmar que a conversa paralela se deu exclusivamente pelo formato das carteiras, porém, logo no segundo momento da atividade já voltamos as duplas para a formação inicialmente pensada (um aluno de frente para o outro), porém, não notamos grandes mudanças no comportamento dos alunos durante as atividades.

As atividades desenvolvidas com os alunos foram exatamente as mesmas desenvolvidas pelas turmas anteriores. As dificuldades em relação à lateralidade e utilização do comando de giro foi semelhante, porém, notamos uma quantidade de conclusão da atividade mais assertiva para essa turma.

4.2.1.3 Aula 3 (21/10/2022) – Atividade plugada (7:20 às 8:05)

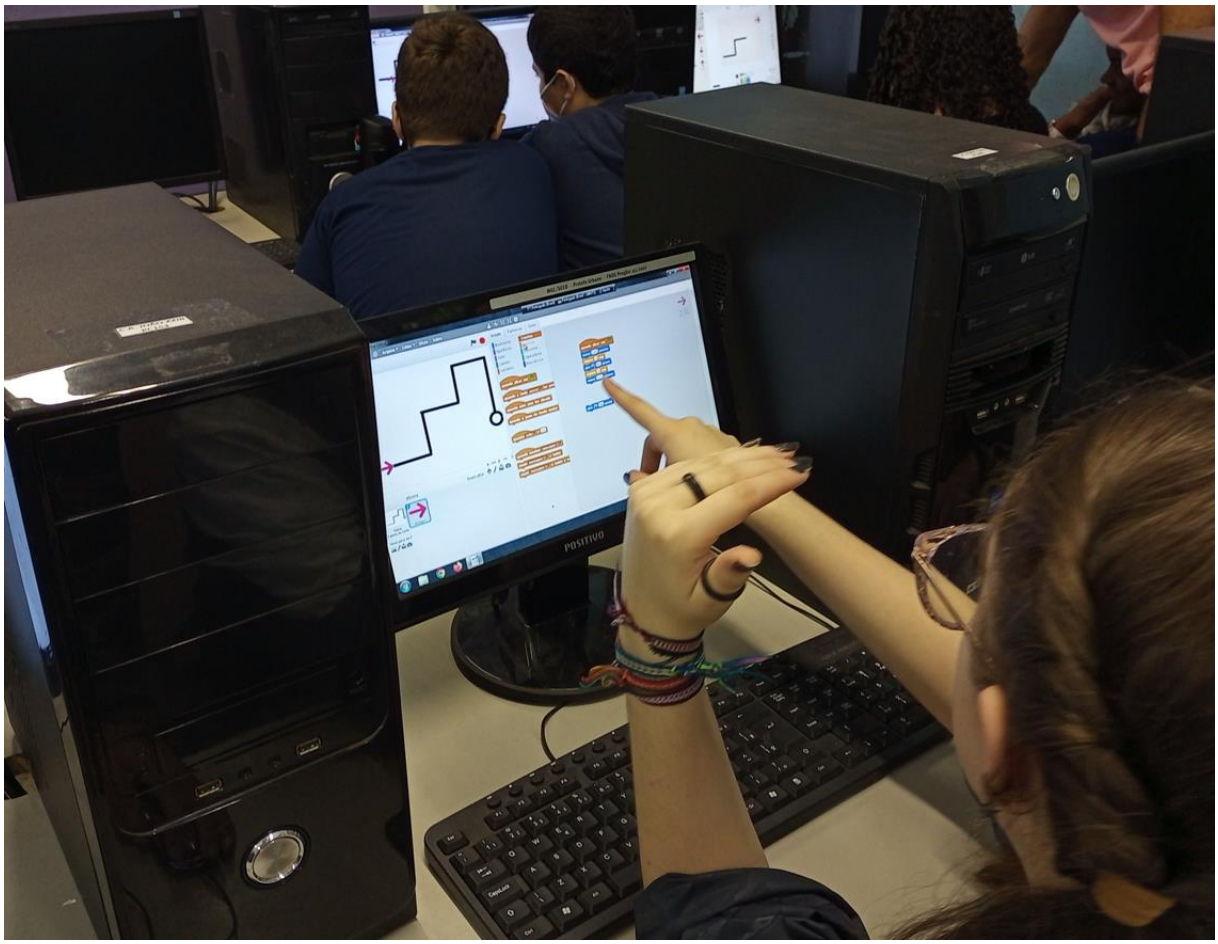
Essa foi a primeira aula em que os alunos tiveram contato com os computadores e optamos por fazer uma mudança considerável em relação às turmas anteriores. Foi possível perceber na primeira aplicação que, devido ao curto tempo total do trabalho, utilizar essa aula com o Blockly Games estava prejudicando o desenvolvimento do projeto final no Scratch. Por esse motivo, optamos por explorar os labirintos no Scratch, de forma semelhante aos apresentados no Blockly Games.

Com essa mudança, os alunos exploraram os mesmos conceitos que propusemos no início, porém, já foram se familiarizando com o Scratch a partir da exploração de comandos de movimento e giro do personagem. Para tal, utilizamos como desafio guiar uma "flecha" até o alvo passando por determinado percurso. Esses percursos foram desenhados pelos próprios alunos. Procurando remeter às atividades desenvolvidas anteriormente em sala de forma desplugada, valorizamos os comandos de girar 90° para direita ou esquerda e o comando de movimento.

A aula toda se deu com os alunos construindo diferentes percursos e programando a flecha para se movimentar pelo caminho. À medida em que os alunos construíam um novo labirinto eles programavam a flecha para percorre-lo e, quando completavam, construíam um novo.

A Figura *50* ilustra o momento em que uma aluna analisa seu código de programação para fazer a flecha percorrer o caminho por ela construído.

Figura 50 - Atividade 1 aula 3



Fonte: o autor

De acordo com o desenvolvimento dos alunos, o pesquisador ia percorrendo as carteiras e conversando com os alunos, procurando instiga-los a avançar em seus trabalhos. O diálogo transcrito a seguir apresenta o pesquisador observando uma das duplas resolvendo o desafio por eles construído e discutindo pequenos ajustes para finalizarem o labirinto.

Pesquisador: Deixa eu ver como é que vocês fizeram a flechinha, vai (pesquisador solicita aos alunos para executarem seu programa) *(SIC)*

[3 segundos de silêncio]

Aluno 1: óh... andou...chegou... (SIC)

Pesquisador: Faltou um pouquinho, né?! Pode subir um pouquinho mais, óh (SIC)

Aluno 2: 130 [aluno sugere nova quantidade de "passos" para a flechinha] *(SIC)*

Pesquisador: Tá vendo, 120 faltou, aí tem que aumentar... não, não, esse aí no último, isso aí. Ajusta e recomeça pra testar. (SIC)

A aula foi finalizada com a maior parte dos alunos conseguindo construir pelo menos 2 labirintos distintos. Algumas duplas conseguiram fazer mais do que isso.

4.2.1.4 Aula 4 (21/10/2022) – Atividade desplugada (8:05 às 8:55)

Nessa aula foram trabalhadas atividades desplugadas em duplas. As duas primeiras atividades de guiar as flechas até o alvo foram idênticas às trabalhadas na primeira intervenção. Já a segunda atividade, da lagarta comilona, sofreu um ajuste no formato em que os alunos deveriam apresentar seu raciocínio.

Ao invés de pedirmos que utilizassem setas para dar a solução às questões, solicitamos que utilizassem os códigos do Scratch conforme ilustrado a seguir.

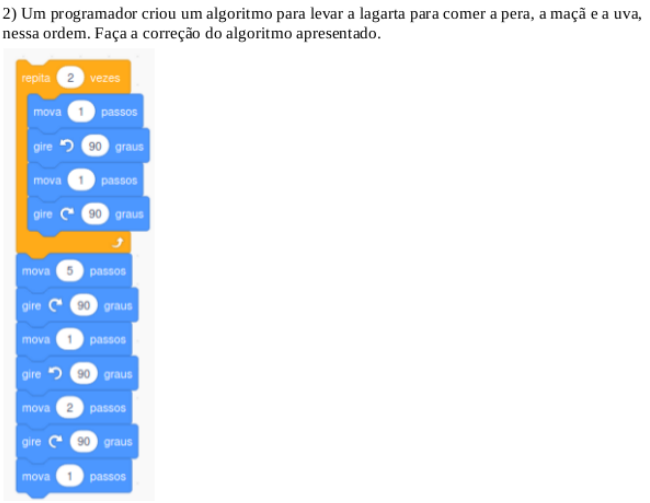
Figura 51 - Atividade 3.1 aula 4



Fonte: o autor

Os desafios da primeira questão (guiar a lagarta até as frutas) foram os mesmos utilizados anteriormente. Já a segunda questão na qual deveriam encontrar uma "falha" no código e corrigir, trocamos as setas por códigos Scratch conforme ilustrado na Figura 52.

Figura 52 - Atividade 3.2 aula 4



Fonte: o autor

Essa mudança no formato de apresentação das atividades objetivou dar uma sequência na utilização dos códigos Scratch, visto que os alunos já haviam utilizado tais comandos nas aulas anteriores.

4.2.1.5 Aula 5 (26/10/2022) – Atividade plugada (10:00 às 10:50)

Essa foi a primeira aula destinada ao projeto COVID-19. Assim como na primeira aplicação das atividades, procuramos primeiramente construir um programa para calcular a média aritmética entre dois números para que os alunos pudessem conhecer os operadores matemáticos e as variáveis. Como os alunos já tinham tido contato inicial com o Scratch em momentos anteriores, não foi necessário utilizar toda a aula para a construção desse programa, apenas os 30 minutos iniciais.

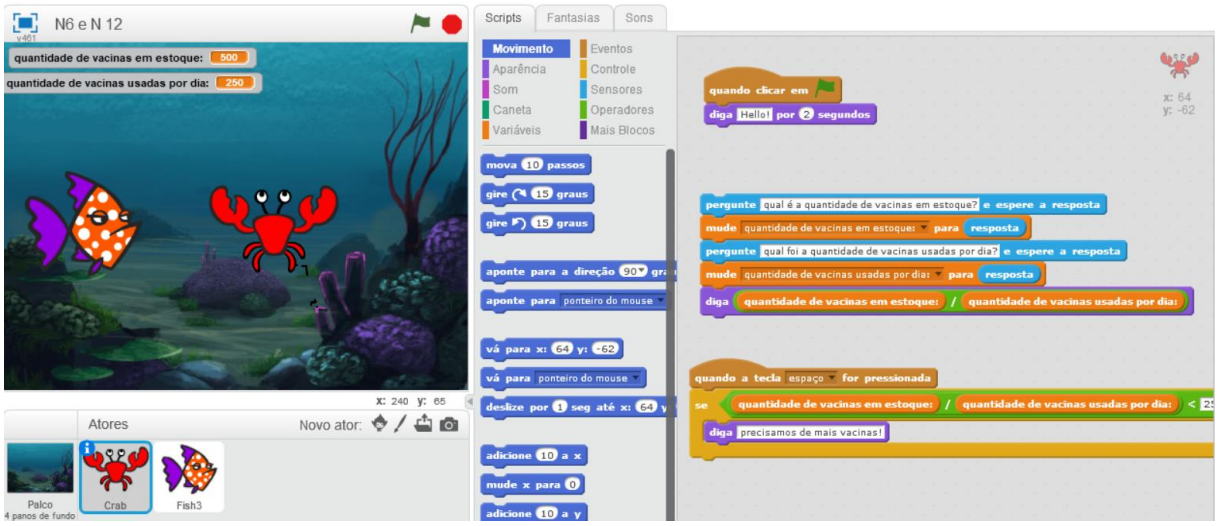
Logo após esse momento já foi iniciado o projeto COVID-19 que, para essa aplicação, foi desenvolvido em duplas e não mais em grupos de 5 ou 6 integrantes. Essa nova formação para o desenvolvimento do projeto funcionou melhor do que para grupos maiores. Foi possível perceber mais envolvimento dos alunos com o trabalho visto que, em duplas, era mais fácil irem

revezando a manipulação do computador, diferente do que ocorria quando estavam em grupos maiores.

4.2.1.6 Aula 6 (26/10/2022) – Atividade plugada (10:50 às 11:40)

Essa segunda aula referente ao desenvolvimento do projeto COVID aconteceu na sequência da primeira, assim não houve perda de tempo até que os alunos chegassem na sala de informática. Os alunos puderam avançar no desenvolvimento da proposta. Ao final dessa aula vários projetos estavam bem avançados, conforme Figura *53*.

Figura 53 - Projeto Scratch aula 6



Fonte: o autor

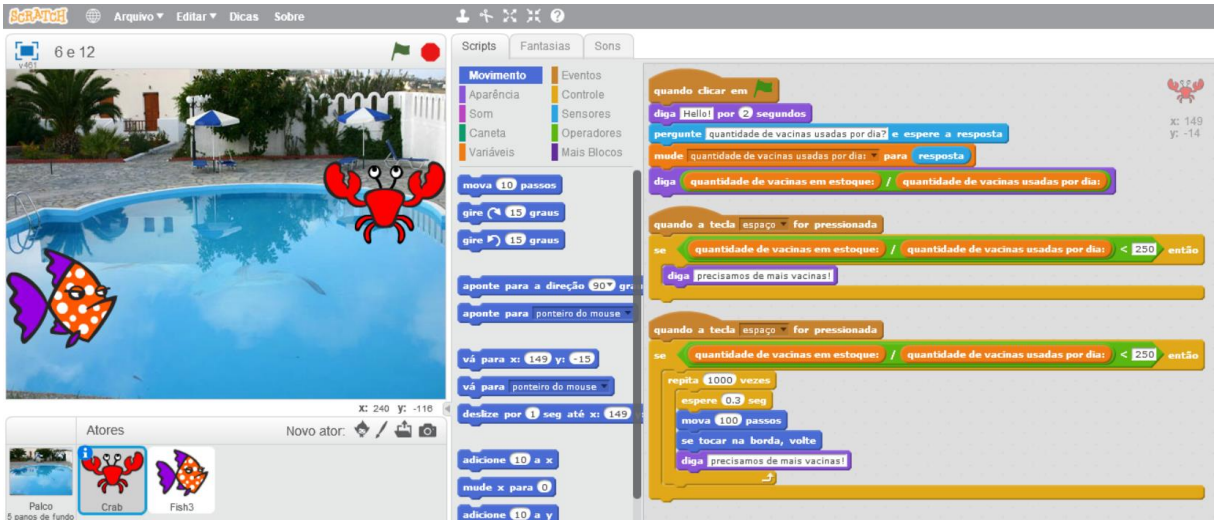
Nesse projeto é possível observar que os alunos conseguiram fazer a inserção de outros personagens, ajustar o plano de fundo, utilizar a programação com variáveis, operador matemático e operador de comparação e condição.

4.2.1.7 Aula 7 (28/10/2022) – Atividade plugada (07:20 às 08:05)

Essa foi a última aula que os alunos tiveram para finalizar seus trabalhos. Assim como na aplicação anterior, houve uma pequena perda de tempo no início da aula até que os projetos fossem novamente distribuídos por meio de *pen drive*, para os alunos darem sequência ao

trabalho. Grande parte das equipes conseguiu avançar na proposta e desenvolver, além da resolução da questão matemática, a parte referente à movimentação dos personagens na tela, conforme apresentamos o projeto seguinte.

Figura 54 - Projeto Scratch aula 7



Fonte: o autor

O código apresentado é referente ao personagem caranguejo, porém, o personagem peixe também possui um código que permite que eles conversem e se movimentem.

4.2.1.8 Aula 8 (28/10/2022) – Apresentação final (08:05 às 8:55)

Assim como nas turmas anteriores nessa última aula os alunos apresentaram seus trabalhos e depois procedemos a uma entrevista com toda a turma. Para essa turma precisamos fazer um pequeno ajuste no formato de apresentação dos trabalhos devido ao curto tempo de aula.

Como a aula 7 também aconteceu nesse dia, os alunos já estavam na sala de informática quando a aula 8 iniciou, desse modo, até que fizéssemos o encerramento da sétima aula para depois nos dirigirmos para a sala de aula, ligar o equipamento multimídia e projetar os trabalhos dos alunos para apresentarem, perderíamos tempo demais e isso prejudicaria a entrevista coletiva no final do trabalho. Dessa forma, solicitamos aos alunos que apresentassem seus projetos nos próprios computadores na sala de informática. O pesquisador foi percorrendo cada um dos computadores e os demais colegas se aglomeravam ao lado do computador para

participar da apresentação. Depois que as apresentações se encerraram fomos para a sala de aula e, nos 12 minutos finais da aula, procedemos à entrevista coletiva.

Foi questionado aos alunos novamente sobre o que seria o PC e, em linhas gerais, fizeram relação com a programação de computadores no sentido de ter que programar algo para ser feito, como guiar o personagem do Scratch até determinado local.

Em relação ao que os alunos acharam sobre as atividades plugadas e desplugadas, houve opiniões divergentes, porém, a maioria disse ser mais fácil trabalhar no computador visto que há um *feedback* imediato, enquanto que na sala, ao programar o colega, a linguagem se torna um fator dificultador.

Como sugestão de melhoria da oficina os alunos disseram que seria mais interessante se houvesse mais aulas na sala de informática, ou fora da sala de aula tradicional.

Finalizada a descrição das aplicações da oficina, na seção seguinte passamos a analisar cada um dos dias dessa aplicação em busca de indícios de potencialização do PC a partir do trabalho com conceitos computacionais bem como com os pilares do PC.

4.3 Síntese das aplicações

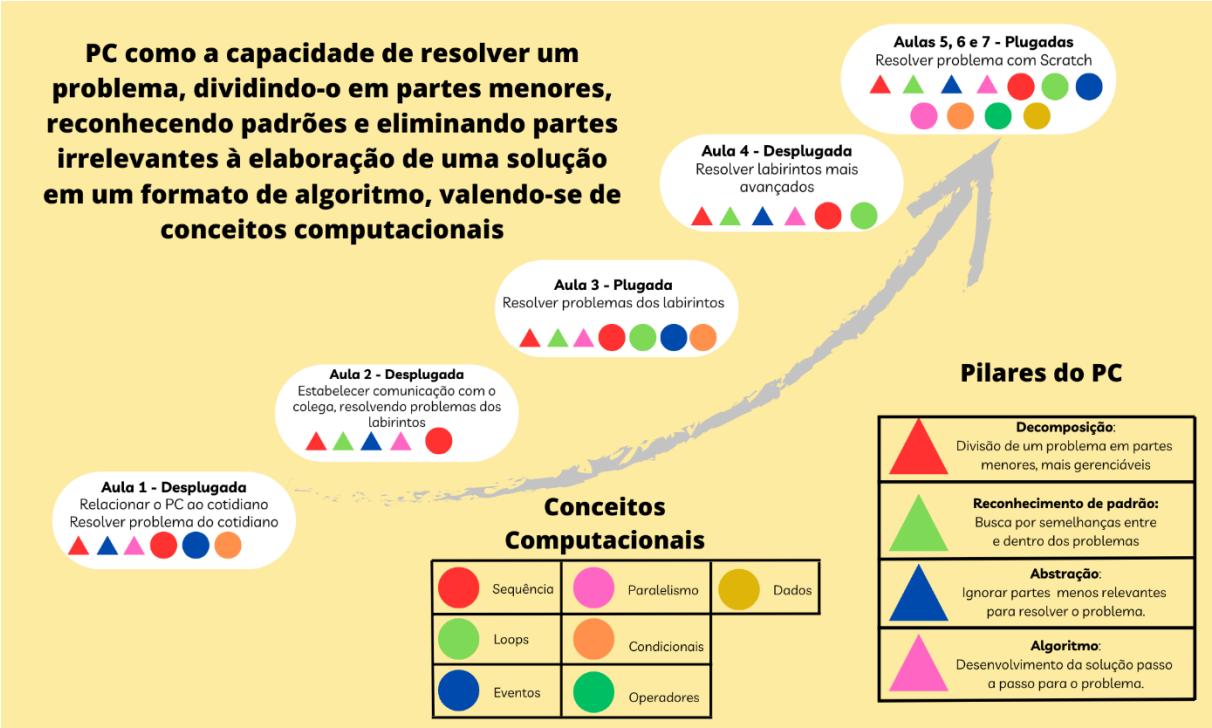
Conforme descrito ao longo do capítulo, a oficina ocorreu em um formato híbrido, procurando trabalhar com atividades que tiveram um nível de dificuldade crescente, intencionando explorar o Pensamento Computacional no formato plugado e desplugado, trabalhando com atividades que estimularam o trabalho individual e coletivo dos alunos.

As atividades desenvolvidas foram elaboradas tendo como base nossa definição para o Pensamento Computacional, buscando tangenciar os quatro pilares do PC (LIUKAS, 2015; BBC BITSIZE, 2015; BRACKMANN, 2017), bem como os sete conceitos computacionais propostos por Brennam e Resnick (2012).

O infográfico apresentado na Figura *55* procura sintetizar a oficina, destacando os pilares e os conceitos explorados em cada uma das aulas, conforme descrevemos ao longo do capítulo.

Na ilustração, apresentamos as aulas como degraus de uma escada, visto que, a cada encontro, o grau de dificuldade das atividades foi sendo incrementado, e os pilares e conceitos computacionais sendo explorados cada vez mais de forma avançada, alternando entre o formato plugado e desplugado. Assim, ao galgar cada degrau dessa escada, os alunos foram, gradualmente, potencializando seu PC.

Figura 55 - Infográfico resumo da oficina de PC



Fonte: o autor

É importante destacar que, tanto os pilares do PC quanto os conceitos computacionais que destacamos para cada uma das aulas, foram aqueles que ficaram mais evidentes ao longo do trabalho, conforme apresentado ao longo das descrições. Outros que porventura não tenham sido destacados no infográfico, podem ter sido também tangenciados ao longo da oficina em menor intensidade.

No capítulo seguinte procedemos a análise dos dados. Nele iremos destacar de maneira mais pontual os pilares e os conceitos explorados nas atividades, destacando as produções e falas dos alunos, articulando com nosso quadro teórico e fazendo importantes destaques que surgiram para além do PC durante a oficina.

5 ANÁLISE E DISCUSSÃO DOS DADOS

Antes de iniciarmos a análise especificamente dos dados, retomamos nossa questão de pesquisa procurando elucidar o que ela interroga. *Como o Pensamento Computacional pode ser potencializado a partir de uma oficina no formato híbrido (mesclando abordagens plugadas, desplugadas e desenvolvimento de projeto)?*

Ao iniciarmos nossa interrogação questionando "como o Pensamento Computacional pode ser potencializado", estamos afirmando que, em alguma medida, esse pensamento já existe nas pessoas, sendo ele muitas vezes aplicado no dia a dia. Sem saberem desse tipo de pensamento, em alguns momentos, as pessoas pensam de forma estruturada, dividem um problema maior em outros menores para resolvê-lo, observam padrões e aplicam a outros casos.

A oficina de PC por nós analisada consistiu em dar luz a essa capacidade de pensamento, valorizando a estruturação e elaboração de uma sequência de passos para resolver os problemas, procurando meios para adaptar esse tipo de pensamento cotidiano em um formato mais estruturado e articulado às tecnologias digitais, perpassando por atividades desplugadas e com as tecnologias digitais.

Quando questionamos como o PC pode ser potencializado pela oficina híbrida, voltamos nossa atenção a essa oficina, olhando especificamente para as atividades plugadas, para as desplugadas e para a articulação de ambas, abrindo oportunidade para discutirmos nuances que surgiram nesse ambiente que podem potencializar o Pensamento Computacional. Para além dos conceitos computacionais e dos pilares do Pensamento Computacional, outras questões surgiram a partir da oficina e serão apresentadas mais adiante.

Apontamos, em nossa metodologia, que faríamos um estudo de caso, entendendo a oficina híbrida como o caso a ser estudado. Ao planejarmos cada momento da oficina, tínhamos objetivos específicos para cada atividade, porém, nem todos foram alcançados. Faremos nossa análise e discussão nos baseando na expectativa (aquilo que pretendíamos ao planejar, ou seja, a perspectiva do professor) e na realidade (aquilo que os alunos fizeram e alcançaram a partir de nossa proposta, ou seja, a perspectiva do aluno). Essa discussão será feita aula a aula, com exceção das aulas 5, 6 e 7 que correspondem ao desenvolvimento do projeto Scratch e, por isso, serão discutidas em uma única seção. A aula 8, referente a apresentação dos projetos e entrevista, servirá de auxílio para discussão de todas as aulas, por isso não haverá uma seção exclusiva para ela.

Conforme apontamos na metodologia, iremos deixar destacado em itálico os conceitos computacionais, os pilares do PC e termos relacionados a ele à medida em que formos analisando cada aula, objetivando trazer mais ênfase a esses pontos para facilitar nossa discussão e entendimento em relação ao tema proposto.

5.1 Aula 1

O objetivo para essa aula foi introduzir o Pensamento Computacional, procurando mostrar que esse tipo de pensamento pode estar presente no cotidiano dos alunos, em simples tarefas do dia a dia. Para isso, iniciamos com uma discussão sobre o assunto que foi conduzida por meio de uma operação matemática mostrada na lousa e um vídeo de um robô aspirador. A partir disso pudemos explorar a ideia inicial do que é um *algoritmo* e de como *resolver um problema* por meio de uma *sequência* de passos.

Durante a apresentação e discussão sobre o vídeo do robô aspirador, falas como "professor, eu posso entender o *algoritmo* dele", "*se* ele encontrar obstáculo, vire para o outro lado e continue andando", "ele foi programado pra quando ele bater ele virar" apresentam indícios de que os alunos, ao se envolverem na discussão sobre o robô aspirador e conjecturarem sobre a programação existente nele, exercitaram a ideia de *algoritmo* e possivelmente tiveram esse pilar do PC potencializado, na medida em que discutiram sobre a *resolução do problema* do robô que era aspirar, imaginando sobre um possível *algoritmo*.

A segunda parte da aula consistiu em solicitar aos alunos que pensassem em alguma tarefa cotidiana, e descrevessem essa tarefa de forma detalhada, por meio de uma *sequência* de passos a serem seguidos, formando um *algoritmo*.

A expectativa para essa aula era de que os alunos fossem capazes de compreender que as tarefas cotidianas podem ser entendidas como *problemas a serem resolvidos* por meio de passos e, nesse formato, sendo possível a construção de um *algoritmo* que resolva esse problema.

Nossa atividade se assemelhou ao que Ortiz e Raabe (2016) propuseram ao desenvolver uma oficina de Pensamento Computacional com alunos da EJA, solicitando que descrevessem uma atividade cotidiana, como trocar pneu de carro ou fazer um bolo, instigando os alunos a pensarem de forma estruturada e descrevendo essa atividade em um formato que todos pudessem entender e executar a tarefa de uma mesma forma, estimulando o entendimento do algoritmo. Marquiori (2021) também fez algo semelhante ao explorar um pouco do Pensamento Computacional em uma oficina para alunas por meio da construção de receitas de bolo.

Assim como em outros casos da literatura, a maior parte dos estudantes que participou de nossa oficina descreveu atividades cotidianas no formato de *algoritmo*. Dos 29 alunos do 7A que participaram dessa primeira atividade, 21 descreveram tarefas cotidianas utilizando *sequência* de passos conforme era esperado. Algumas dessas tarefas foram limpar cocô de cachorro, lavar louça ou fazer exercícios de matemática, que certamente retratam o cotidiano desses estudantes. Outros 3 alunos dessa turma parecem não ter compreendido bem a proposta, visto que escreveram tarefas aleatórias, sem ter o foco em algo e detalhar passo a passo. Já os outros 5 alunos, utilizando passo a passo, descreveram um robô companheiro, que pudesse ouvir e opinar sobre o que foi dito. Um dos alunos chegou a mencionar que seu robô era um psicólogo.

Esse último grupo de alunos chamou a atenção, visto que seus trabalhos não estavam relacionados a uma determinada tarefa conforme havíamos solicitado, mas se relacionavam a outros fatores que certamente estavam latentes a eles. A partir de uma consulta feita aos professores da turma, descobrimos que no início do ano que fizemos a coleta de dados, os alunos haviam tido problemas entre si relacionados a críticas em um grupo de mensagens instantâneas. A escola tomando parte do ocorrido acionou o núcleo de psicologia que, no mês de outubro, procedeu a uma intervenção na classe, conversando com os alunos sobre o assunto. Essa conversa ocorreu exatamente no mesmo mês em que fizemos a coleta de dados e, provavelmente, influenciou na decisão dos alunos em escrever sobre isso na primeira atividade que propusemos.

Souza *et al*. (2020) destacam que fatores externos, como os emocionais, influenciam no contexto escolar no que tange ao aprendizado. Conforme os autores, o conflito entre pares, como *bullyng*, influencia diretamente no desempenho escolar dos estudantes. As atividades apresentadas por esses 5 alunos nesse primeiro momento de nossa oficina corroboram o fato de que os fatores externos operam ativamente no desempenho escolar. Esses alunos estavam "incomodados" com outras questões que estão além da escola e, em nossa atividade, tiveram abertura para de alguma forma expressar esse incômodo, mesmo que aquilo não estivesse exatamente relacionado a proposta.

Dessa forma, destacamos que conseguimos na primeira aula da oficina discutir com os alunos sobre o Pensamento Computacional, especialmente sobre *resolver um problema* por meio de uma *sequência* de passos, formando um *algoritmo*, e apresentar a eles algumas relações desse estilo de pensamento com o cotidiano, sendo que a maior parte dos alunos pareceu entender essa relação a partir das atividades desenvolvidas. Nessa aula também pudemos observar como os fatores externos ao ambiente escolar operam fortemente em sala de aula,

sendo uma variável muitas vezes desconsiderada, porém, observamos que ela possui grande potencial de influência no meio escolar.

5.2 Aula 2

Na segunda aula foram trabalhadas quatro atividades desplugadas, nas quais os alunos deveriam programar o colega a partir de caminhos desenhados em folhas de papel. Essa atividade procurou explorar, especialmente, o pilar *algoritmo* do Pensamento Computacional, mas também *divisão de problemas*, *abstração*, *reconhecimento de padrões* e o conceito computacional de *sequências*.

A utilização de setas para representar o percurso foi inspirada em atividades que Brackmann (2017) utilizou em seu trabalho, e que Liukas (2015) utilizou nos desafios de seu livro. A expectativa com essa atividade era que os alunos, ao assumirem o papel de programador, observassem o quadriculado com o caminho e então buscassem estabelecer um formato de comunicação com o colega computador para informar a ele aquele caminho. O colega computador, utilizando as setas, deveria construir o *algoritmo* e então reproduzir no seu papel quadriculado o mesmo percurso que o programador possuía em sua folha.

Nosso intuito com a proposta, além de explorar *sequências, algoritmos, divisão de problemas, abstração e reconhecimento de padrões*, se relacionava também com o estabelecimento de uma linguagem de comunicação entre os alunos. Essa tentativa de estabelecimento de comunicação entre os alunos por meio de uma "nova" linguagem poderia contribuir para eles desenvolverem a capacidade de traduzir uma determinada situação em outra linguagem, conforme é defendido pela BNCC (BRASIL, 2018).

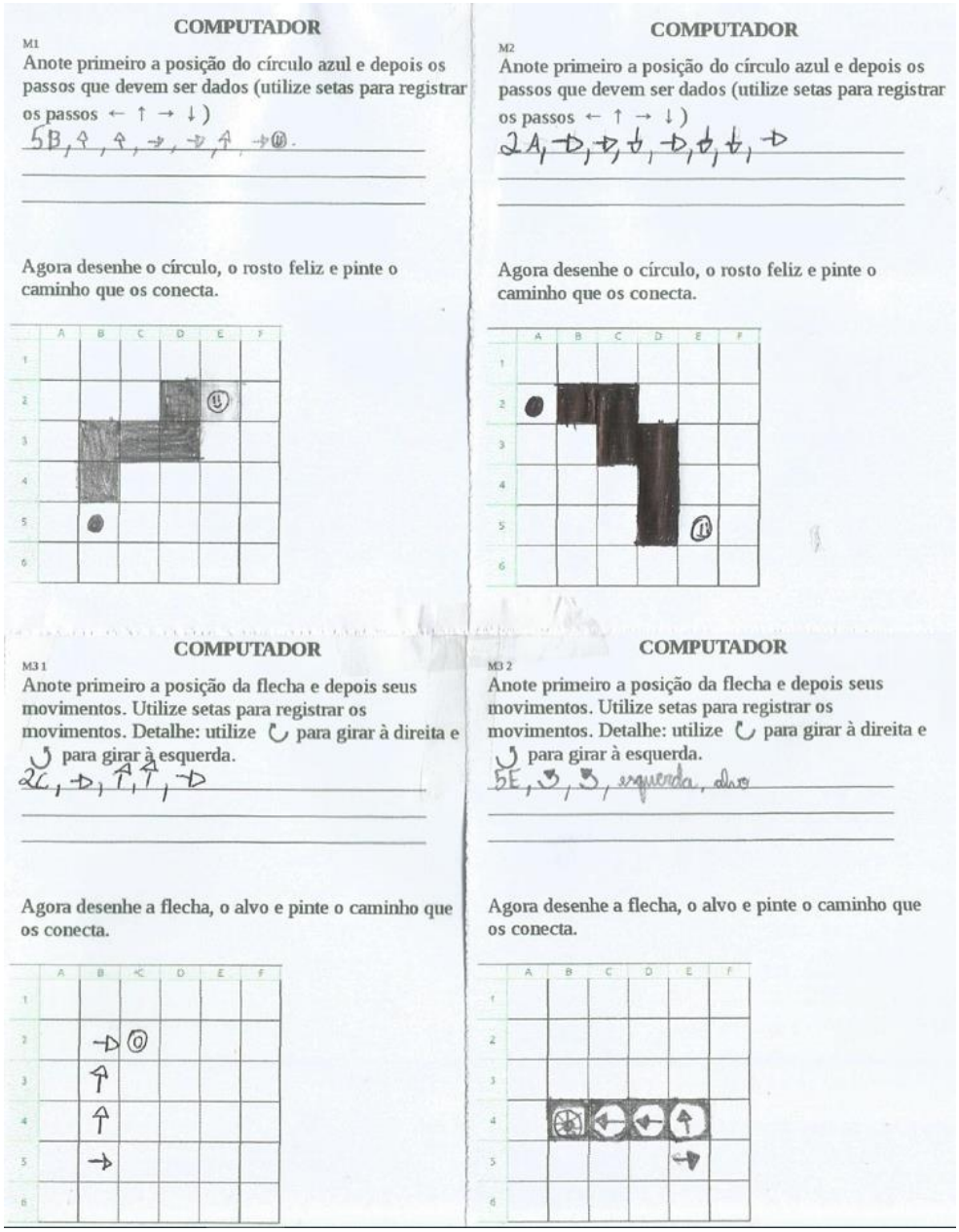
Essa nova linguagem trouxe diversas confusões, não só na parte da comunicação, mas também na parte do entendimento da dupla para anotar a informação recebida. Transcrevemos o diálogo entre o pesquisador e duas alunas que estavam com dificuldade de se comunicarem. O pesquisador as acompanhou durante toda a resolução da primeira atividade dessa aula, orientando como deveria ser estabelecida a comunicação entre elas e como o "computador" deveria ir registrando as instruções recebidas pela colega "programadora".

Pesquisador: Diga, oi (SIC)
Aluna 1: A gente que tá te chamando. Professor, ela não entendeu como é que...
Pesquisador: Como é que cê falou com ela, me diga. Mas não é pra você olhar, você só vai desenhar (SIC)
Aluna 2: a bolinha tá na quinta fileira, no B. (SIC)
Pesquisador: Ahn... (SIC)

Aluna 1: ela tá perguntando... (SIC)
Aluna 2: Aí eu tenho que fazer um círculo ou a seta? (SIC)
*Pesquisador: Não, círculo! Olha aqui, óh. Olha aqui, olha só o que que tá dizendo aqui em cima, óh. Aqui cê vai colocar onde tá a bolinha, que ela já falou, na 5B. Então você vai escrever aqui, 5B. Você já sabe que a bolinha tá lá. Depois você vai dizendo pra ela o caminho (*pesquisador fala para a aluna 1*). Como é que você vai dizer o caminho até chegar no rostinho feliz? (SIC)*
Aluna 1: Ahn, aí depois você vai andar duas, dois quadradinhos... (SIC)
Pesquisador: pra onde? (SIC)
Aluna 2: pra onde? pra cima? Pra baixo? (SIC)
Aluna 1: pra cima! (SIC)
Pesquisador: pra cima, dois pra cima. Então duas setinhas pra cima. Depois... (SIC)
Aluna 1: aí depois... você vai andar duas casas a direita(SIC)
*Pesquisador: duas setinhas pra direita. Vai fazendo! (*pesquisador alerta aluna 2*) Duas seti... aqui, aqui, óh, setinhas, duas setinhas pra cima... isso, 5B, depois duas setinhas pra cima, depois, duas pra direita que ela falou... Que mais? (SIC)*
Aluna 1: aí depois você sobe uma setinha pra cima(SIC)
Aluna 2: hum? (SIC)
Aluna 1: depois você sobe uma setinha pra cima. E depois cê... (SIC)
Aluna 2: duas? (SIC)
Aluna 1: Uma! (SIC)
Pesquisador: Continua! (SIC)
Aluna 1: e depois você... vai à direita e chegou(SIC)
Pesquisador: uma pra direita e chegou(SIC)
Aluna 1: aí no final você chega no 2E(SIC)
Pesquisador: Agora você usando essas setinhas, desenha aqui o caminho(SIC)

Esse diálogo é referente a primeira das 4 atividades desplugadas que foram aplicadas na segunda aula. Abaixo ilustramos essas 4 atividades desenvolvidas pela dupla referente ao diálogo.

Figura 56 - Atividades desplugadas - dia 2, dupla 1



Fonte: o autor

A primeira imagem do canto superior esquerdo (M1) é referente ao primeiro momento dessa atividade. O auxílio do pesquisador às alunas se deu exatamente na resolução dessa atividade. É possível observar que elas conseguiram concluir com êxito o que foi proposto, desenhando perfeitamente os pontos de início e fim do percurso bem como escrever o percurso por meio das setas (*algoritmo*). Já o segundo momento (M2), muito semelhante ao primeiro, porém com as alunas trocando de papéis (computador vira programador e vice-versa), as alunas trabalharam de maneira independente e conseguiram concluir com êxito tanto o *algoritmo* com as setas na *sequência* necessária, quanto o percurso desenhado.

Já as duas partes do terceiro momento (M3 1 e M3 2), as alunas fizeram confusão ao anotar o *algoritmo* com as setas. O primeiro momento sequer aparece a seta para indicar o giro, já no segundo momento aparecem duas, mas que resultariam em outro caminho distinto daquele por elas apresentado.

Ao que se refere a primeira etapa dessa atividade (M1 e M2), inferimos que as alunas parecem ter compreendido o formato de representação por meio das setas, visto que conseguiram, de forma independente, resolver o segundo desafio escrevendo corretamente a *sequência* de setas formando o *algoritmo* necessário para a construção do caminho. Dessa forma inferimos que, possivelmente, houve *potencialização do PC*, pelo menos no que tange ao desenvolvimento de *sequência* e os *algoritmos* com as setas. Já na parte final da atividade (terceiro momento), com a inserção um novo comando para efetuar o giro da flecha, houve confusão por parte das alunas e elas não conseguiram compreender bem o funcionamento desse giro.

O mais intrigante ao observar esse terceiro momento é: como o percurso no quadriculado está correto sendo que o algoritmo com as setas está errado? A primeira resposta que vem à mente a partir dessa pergunta é apenas uma: colaram. Pois bem, mas porque fizeram isso se a atividade, conforme foi claramente explicado para os alunos, era de participação facultativa e não tinha nenhum valor de nota para os alunos?

A hipótese que levantamos para isso é que o sistema educacional tradicional está arraigado e, por esse motivo, as alunas estavam preocupadas em apresentar um resultado "correto", visto que nesse formato de ensino é importante apresentar uma resposta certa para conquistar pontos e ser promovido de ano escolar. Leão (1999) ressalta que nesse formato o professor é o detentor do conhecimento e os alunos procuram ouvi-lo e resolver atividades corretamente.

Conforme afirma o professor Marco Braga, a escola na realidade é um grande game no qual os alunos precisam conquistar uma determinada quantidade de pontos para conseguirem passar de fase. Nesse formato a aprendizagem se torna apenas uma consequência, pois a preocupação maior é como fazer para mudar de fase (GEPETEC UNIRIO, 2023).

Conjecturamos que dessa forma as alunas queriam trazer uma resolução que "agradasse" ao professor pois, no ensino tradicional é ele que tem o "poder" de dar a nota e dizer se o aluno estará aprovado (irá passar de fase). O caminho mais simples para acertar a atividade era simplesmente olhar o "gabarito" que estava nas mãos da colega, porém, como nesse gabarito só havia o caminho a ser percorrido sem as setas que formavam o algoritmo, aconteceu que o

algoritmo criado pelas alunas não estava condizente com o caminho. Tal fato aconteceu não apenas com a dupla analisada na presente discussão, mas em várias outras.

Apesar dessa questão, foi possível perceber que houve exploração do PC por meio da *resolução de um problema* (guiar objeto até o alvo) a partir da construção de uma *sequência* de setas formando o *algoritmo* que traçava o caminho. A comunicação entre os alunos foi um fator complicador, visto que não havia uma regra de como se comunicar e, portanto, os alunos deveriam estabelecer um formato para essa comunicação. Conforme relatado pelos próprios alunos na entrevista final da atividade, essa foi uma grande dificuldade.

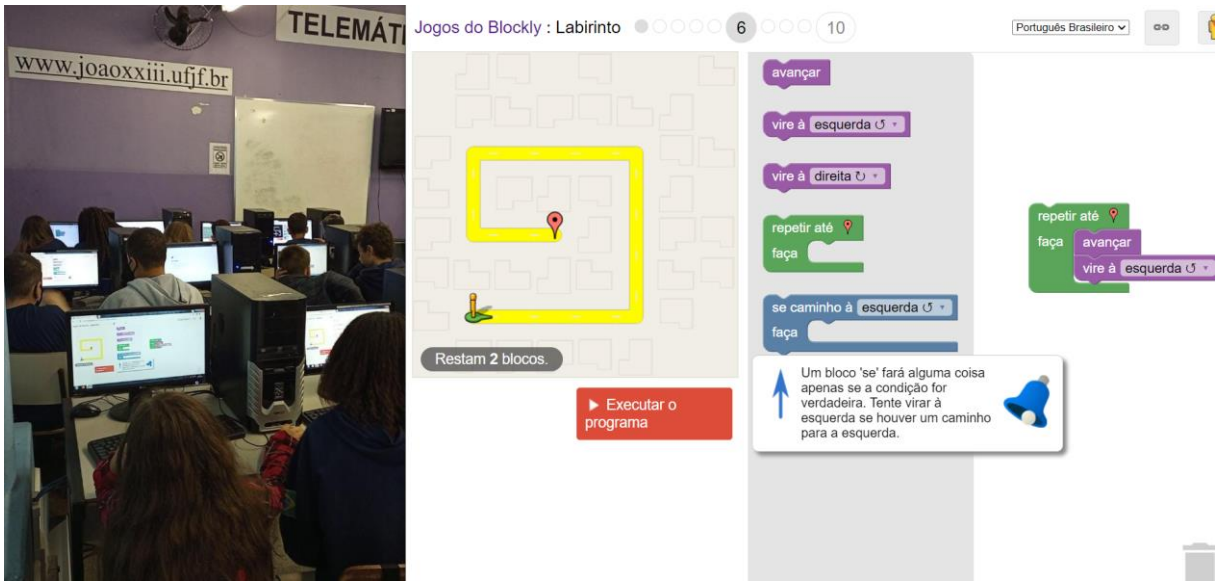
5.3 Aula 3

A terceira aula de nossa intervenção aconteceu na sala de informática com atividades plugadas. Na primeira aplicação (turmas B e C) trabalhamos os labirintos do jogo Blockly Games. Esse jogo permitiu aos alunos explorar o mesmo estilo de *resolução de problemas* trabalhado em sala de forma desplugada, porém, com mais recursos e complexidade. Durante essa aula os alunos puderam explorar três dos sete conceitos computacionais destacados por Brennam e Resnick (2012): *sequência*, *loops* e *condicionais*. Além dos conceitos computacionais, os alunos também tiveram o primeiro contato com a construção de *algoritmos* por meio da linguagem de programação em blocos.

Já os alunos participantes da segunda aplicação (turma A) não tiveram contato com o Blockly Games nessa aula, mas sim com o Scratch. Nas turmas participantes da primeira aplicação (7B e C), observamos que o tempo planejado para trabalhar com o projeto Scratch durante as aulas 5, 6 e 7 foi curto, visto que a primeira dessas aulas foi utilizada para conhecer o *software*, assim, para a segunda aplicação (7A), ajustamos a exploração dos mesmos conceitos computacionais diretamente no Scratch, com os alunos construindo labirintos e programando flechas para atingirem alvos.

À esquerda da Figura *57* estão alunas jogando a sexta fase do jogo labirinto. À direita da mesma figura é apresentado um zoom da tela do computador dessas alunas.

Figura 57 - Alunas jogando fase 6 Blockly Games

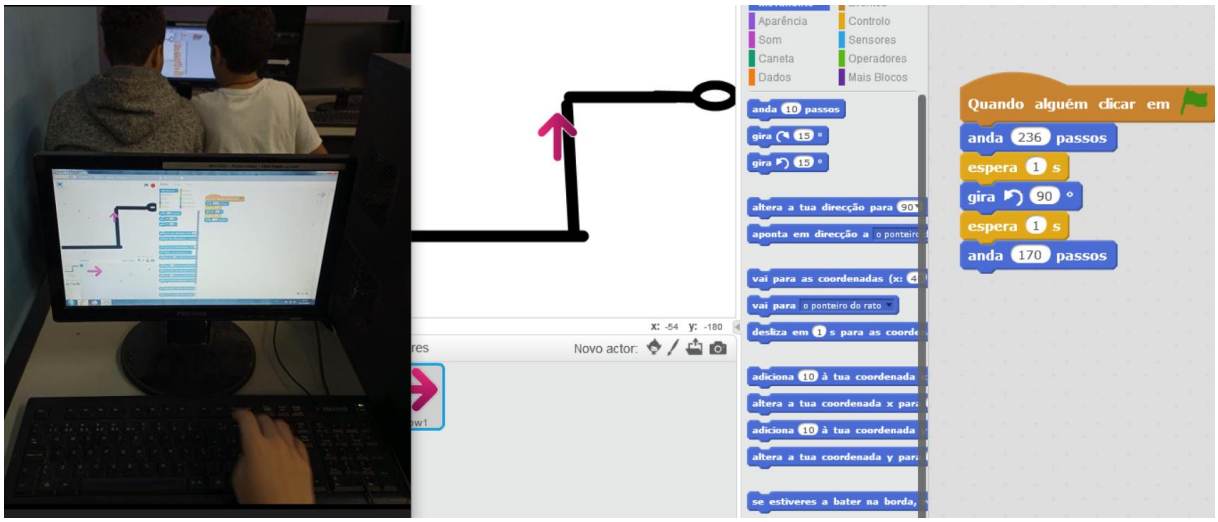


Fonte: o autor

Conforme dissemos, é possível observar nessa fase do jogo os alunos trabalhando com três dos conceitos computacionais propostos por Brennam e Resnick (2012): *sequência* (ao empilharem os blocos em uma determinada ordem para executar o movimento pretendido); *loop* (bloco verde que indica uma repetição dos blocos internos a ele) e *condicionais* (bloco azul que executará uma ação se for atendido o requisito de haver caminho). À medida que os alunos vão avançando de fase, novos conceitos computacionais vão sendo exigidos, além da complexidade dos labirintos irem aumentando. A manipulação desses blocos e a construção de *algoritmos* para *resolver o problema* de guiar o boneco até o alvo nos leva a crer que houve ali uma potencialização do PC dos alunos.

Já na segunda aplicação, os alunos, utilizando Scratch, construíram seus próprios labirintos e programaram uma flecha para ir percorrendo o caminho. À esquerda da Figura *58* ilustramos uma dupla programando a flecha para percorrer o caminho por eles construído e, ao lado direito um zoom do código de programação.

Figura 58 - Alunos construindo labirinto Scratch



Fonte: o autor

Diferentemente dos alunos que trabalharam com o Blockly Games, no código desenvolvido por essa dupla não há blocos de condicionais ou repetição, porém, é possível observar que eles exploraram os conceitos de *sequência* (ao empilharem os blocos em uma determinada ordem para executar o movimento pretendido) e também de *eventos* (uma coisa faz com que outra aconteça, ou seja, ao clicar na bandeira verde o programa é executado).

Para além dos conceitos computacionais que foram explorados nessa atividade, um fator que merece atenção é o *feedback* instantâneo que as tecnologias digitais proporcionam. Esse ambiente de programação, seja com Blockly Games ou Scratch, possui características peculiares que permitem ao aluno enxergar seu erro em um formato muito particular. Diferentemente do que acontece na sala de aula tradicional, durante a programação de computadores, o erro, que na verdade é um não funcionamento ou um funcionamento distinto daquilo que se desejava, pode desencadear o início da espiral de aprendizagem proposta por Valente (2005a).

Essa espiral de aprendizagem, conforme Valente (2005a) destaca, acontece à medida em que o aluno *descreve* seu programa (transforma em códigos o que pretendia) e então o *executa* observando o resultado e *refletindo* sobre ele. Caso esse resultado não esteja de acordo com o esperado, o aluno *depura* seu código, ou seja, faz os ajustes necessários, executa novamente e observa o resultado. Esse ciclo de ações continua até que o aluno alcance o resultado desejado. A Figura *59* ilustra esse ciclo.

Figura 59 - Ciclo de ações no contexto da programação de computadores



Fonte: Scherer e Fernandes (2014, p. 148)

Esse ciclo de ações aconteceu diversas vezes durante nossa intervenção. Às vezes de forma espontânea, com discussões entre os próprios alunos; às vezes de forma estimulada, com discussões com o pesquisador, como é o caso apresentado a seguir.

O diálogo transcrito abaixo apresenta um momento em que o pesquisador vai até uma dupla de alunas do 7A que estava programando a flecha para percorrer o labirinto por elas construído. Elas executam o programa e veem que a flecha não chega no local em que desejavam e então ajustam o código.

Pesquisador: Óh, gostei desse labirinto, hem?! *(SIC)* [3 segundos de silêncio enquanto as alunas executam o programa]
Aluna 1: ai, quase! (SIC) [aluna observa que a flecha se aproxima do alvo, mas ainda não está conforme gostariam]
Aluna 2: 100 (SIC) [a colega sugere uma nova quantidade de passos a serem ajustados no programa]
[9 segundos se passam, nesse período as alunas conversam baixo, de forma inaudível e reexecutam o programa]
Pesquisador: óh, legal...legal... úh, quase! (SIC)
Aluna 2: 110 (SIC) [aluna sugere novo valor para ajuste do programa]
[3 segundos de silêncio enquanto há nova execução do programa]
Aluna 2: 115! (SIC) [aluna finaliza fazendo um último ajuste para a flecha chegar exatamente no centro do alvo]

A partir desse diálogo é possível observar que a espiral proposta por Valente (2005a) ocorre por 3 vezes. Logo na primeira fala do pesquisador, é possível perceber que o código já está pronto e que acontece a primeira execução do programa. Após a interjeição da aluna 1 expressando que a flecha quase alcançou o alvo, a aluna 2 sugere novo valor, ou seja, naquele momento já houve o *feedback* imediato do Scratch que permitiu a reflexão e a depuração ajustando para "100" a quantidade de passos da flecha. Com a interjeição do pesquisador expressando que "quase" a flecha alcança o alvo, inferimos que houve nova execução do programa e que a aluna, novamente, reflete sobre o resultado e na sequência já sugere o valor 110. O ciclo reinicia e, pelo tempo de silêncio entre as falas é possível perceber que o programa estava sendo novamente executado. Por fim a aluna sugere, em tom de exclamação, o valor 115, ou seja, um mínimo ajuste em relação ao anterior para que a flecha chegue exatamente onde queriam.

Essa questão do *feedback* imediato proporcionado pelas tecnologias digitais, segundo os alunos, torna a atividade mais "fácil" quando comparada com as atividades desplugadas. O fato de ser possível observar a evolução de seu trabalho imediatamente após os ajustes certamente faz a diferença. Tal fato ficou comprovado a partir da entrevista coletiva que aconteceu no final da oficina.

Em determinado momento da entrevista com os alunos do 7A, o pesquisador os relembra sobre as atividades desenvolvidas, procurando trazer à tona a terceira aula em que eles programaram a flechinha no Scratch de forma semelhante ao que tinham feito em sala, de forma desplugada. Feita essa retomada, o pesquisador questiona aos alunos sobre qual das atividades eles sentiram mais dificuldade em executar, se em sala com os colegas (atividade desplugada) ou se na sala de informática (atividade plugada).

No primeiro momento vários alunos falam ao mesmo tempo, alguns afirmando que em sala foi mais fácil e outros afirmando que foi mais difícil. Procurando organizar a discussão, o pesquisador vai questionando a alguns alunos porque para eles a atividade desplugada havia sido mais difícil do que a plugada.

Pesquisador: Porque que aqui na sala você achou mais difícil do que lá? (SIC)

Aluno 1: Porque lá... aqui era diferente, você tem que... falar pra pessoa e daí a gente não conseguia entender... (SIC)

Pesquisador: Mas lá no computador você tinha que falar pro computador(SIC)

Aluno 1: Não! Lá é um comando... não é falar tipo... (SIC) [outro colega o interrompe para complementar sua fala]

Aluno 2: É porque aqui a nossa dupla é um ser humano que é um... (SIC) [aluno aponta para a própria cabeça] *outro animal! (SIC)* [colegas falam junto] *A gente falava e ele não entendia*
[muita discussão e falas ao mesmo tempo]

Pesquisador reorganiza a discussão e solicita que os alunos façam silêncio, passando a palavra para uma aluna no fundo da sala que estava com a mão erguida para falar.

*Aluna 1: Lá no computador a gente mandava fazer alguma coisa, e tipo... já dava pra ver mais ou menos como ficava. Aí aqui não, tipo... ela [*aponta para a colega*] podia entender de outra forma [*aluna gesticula com os braços representando uma confusão*], lá no computador era só aqueles comandos mesmo. (SIC)*

*Pesquisador: Então você achou mais difícil aqui em sala. Legal... Olha só que interessante a ideia da colega [*pesquisador chama atenção de toda a turma para a fala da aluna*]. Porque lá no computador, você dava o comando e você conseguia ver o que tava acontecendo... aqui não. (SIC)*

Depois dessa fala do pesquisador outro aluno pede a palavra e concorda com ele, dizendo que achou mais fácil na sala de informática por ver o retorno e, por fim, outro aluno pede a palavra para dar sua opinião.

*Aluno 3: Então... lá foi mais fácil. Você tem essa facilidade de usar o computador [*inaudível*] aí, lá você só pegar o comando e colocar, aqui você tem que falar o comando, pode ser que pessoa não entende, entende errado, ela pode fazer uma coisa que você não falou (SIC) [*inaudível*]*

Pesquisador concorda com o aluno e esse assunto é encerrado.

A partir das falas dos próprios alunos é possível observar que o retorno imediato proporcionado pelo computador é um grande diferencial da atividade plugada. É possível inferir que esse formato faz oposição à sala de aula tradicional no quesito de avaliação. No contexto da programação de computadores, muitas vezes o crivo parte do próprio aluno que, ao ver o resultado imediatamente após sua construção, faz ajustes em seu trabalho e verifica o resultado e, dessa forma, vai avançando até alcançar o resultado que ele mesmo deseja.

Na sala de aula tradicional é diferente. Na maior parte das vezes o crivo vem do professor, detentor do conhecimento. Além disso, o retorno do professor não é imediato, demanda um tempo para levar as atividades para casa, fazer a correção e dar o retorno aos alunos. Esse retorno em grande medida vem no formato de uma nota que, muitas vezes, não é

debatida, especialmente se o aluno fez uma pontuação mínima necessária para sua aprovação (CARVALHO, 2018).

Além do *feedback* imediato, outro ponto que merece destaque a partir das falas dos alunos é em relação à linguagem. Estabelecer uma forma para dizer ao colega o percurso a ser seguido foi um grande desafio, especialmente em relação à lateralidade (direita/esquerda), ainda mais quando estavam assentados um de frente para o outro. Conforme os alunos destacaram, havia uma confusão com isso, um dizia uma coisa e o colega entendia outra. Já no computador esse problema não aconteceu. Como os alunos utilizaram programação em blocos, os comandos já estão prontos, ou seja, já existem os blocos para programar, cabendo aos alunos apenas sua utilização de forma estruturada para que consigam alcançar o objetivo proposto.

Concluímos que nessa aula o PC pode ter sido potencializado na medida em que os alunos, no computador, manipularam a linguagem de programação em blocos e, com ela, construíram *sequências* de comandos formando *algoritmos* para *resolver o problema* de guiar um objeto até seu destino. Além disso, a partir da espiral de aprendizagem de Valente (2005a), foi possível perceber uma constante reflexão e depuração em relação aos *algoritmos* construídos na medida em que recebiam o *feedback* imediato do computador, reforçando a potencialização do PC.

5.4 Aula 4

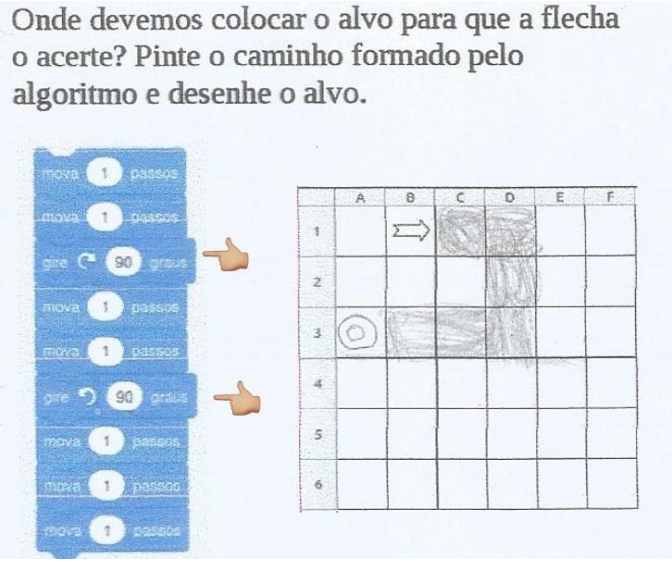
A quarta aula de nossa intervenção aconteceu de forma desplugada, objetivando explorar os conceitos computacionais de *sequências* e *loops* (BRENNAM; RESNICK, 2012) por meio da resolução de labirintos, além dos comandos de giro. Nessa aula priorizamos a representação dos comandos utilizando os blocos de programação semelhantes aos trabalhados na aula anterior com Scratch e *Blockly Games*. O objetivo era que os alunos continuassem a se familiarizar com esse tipo de linguagem e, portanto, nos dois primeiros momentos dessa aula, não utilizamos as setas para indicar o *algoritmo* como feito na aula 2 também desplugada, mas sim os blocos feitos com a linguagem Scratch.

Mesmo com a utilização do *algoritmo* com blocos do Scratch, apenas 5 das 14 duplas da turma 7A conseguiram desenhar o percurso correto da flecha a partir do *algoritmo* dado na primeira atividade; já na segunda atividade, apenas uma dupla desenhou o percurso errado (essa dupla receberá especial atenção em uma seção da tese mais à frente).

É possível inferir que a grande quantidade de erros que aconteceram na primeira atividade se deveu aos comandos de giro, especialmente o segundo comando que, visualmente,

é muito semelhante ao primeiro, mas com o sentido oposto. Os comandos de giro estão apontados na Figura *60*.

Figura 60 - Atividade 1A aula 4

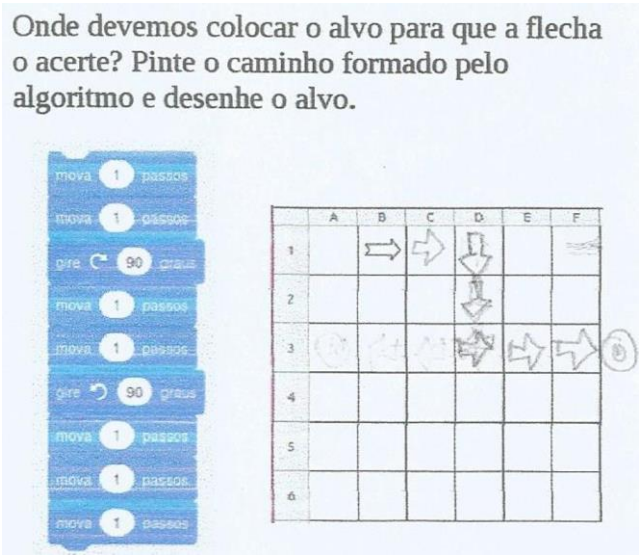


Fonte: o autor

Conforme é possível observar na resolução dessa dupla, houve uma interpretação do segundo giro para o mesmo lado do primeiro, ocasionando em um percurso equivocado, como o que está apresentado.

Durante a atividade, enquanto caminhava entre os alunos e ia observando as resoluções e dando o devido suporte, o pesquisador percebeu que a grande maioria dos alunos estava "lendo" o segundo giro igual ao primeiro e, por esse motivo, deu um alerta para toda turma se atentar a esses comandos, pois eram parecidos, mas não iguais. Mesmo com esse alerta, das 14 duplas participantes, 7 cometeram o erro de forma semelhante ao ilustrado; 4 cometeram outros erros e, as 3 duplas que acertaram a atividade fizeram a correção após o alerta, conforme é possível observar pelo sombreado apagado na Figura *61*.

Figura 61 - Atividade 1B aula 4



Fonte: o autor

Conforme já discutido anteriormente, esse erro provavelmente não aconteceria, ou pelo menos não ficaria "registrado" se essa atividade tivesse sido desenvolvida com suporte do computador, visto que os alunos teriam um retorno imediato daquilo que estavam fazendo, ou seja, na tela do computador eles poderiam inserir os códigos que foram apresentados de forma impressa e, ao executarem, veriam o resultado "diferente" daquele por eles desenhado e rapidamente fariam a correção.

Para além da discussão já tecida anteriormente em torno do *feedback* imediato que as tecnologias digitais proporcionam, podemos inferir que essa dificuldade pode ter ocorrido devido à pouca familiaridade dos alunos com a linguagem de programação em blocos e sua simbologia.

O bloco do Scratch que representa o giro possui uma pequena setinha curva indicando se o giro irá acontecer para a direita ou para a esquerda. No primeiro comando de giro, essa curva da setinha apontava da esquerda para direita, ou seja, a flechinha deveria apontar para baixo, mesma direção da setinha do giro. Já no segundo bloco de giro, a setinha curva apontava da direita para a esquerda, mas como a "pontinha" da setinha estava virada para a esquerda, os alunos apontaram a flechinha para a mesma direção sem se atentarem a quantidade de graus pois, para acontecer o giro conforme foi desenhado por eles, seria necessário um giro de 270º e não apenas 90º.

A expectativa ao preparar essa atividade era que os alunos já tivessem a capacidade de fazer essa interpretação dos comandos de giro, visto que na aula anterior, nas atividades

plugadas, eles haviam trabalhado com esses comandos ao construírem os labirintos no Scratch, porém, ficou claro que apenas uma aula não foi suficiente para que eles pudessem se apropriar dessa linguagem a ponto de serem capazes de, no papel, fazerem a leitura e interpretação desse código.

Segundo o dicionário de filosofia Abbagnano (1998), na realidade o correto seria dizer da "língua" de programação, e não linguagem. A língua é um conjunto organizado de signos linguísticos. Assim, conforme é apontado,

> podem-se distinguir duas espécies de línguas: *históricas,* cuja massa falante é uma comunidade histórica: p. ex. italiano, inglês, francês, etc; 2ª *artificiais,* cuja massa falante é um grupo que se distingue por uma competência específica; são as línguas das técnicas específicas (às vezes chamadas impropriamente de linguagens); p. ex.: língua matemática, língua jurídica, etc (ABBAGNANO, 1998, p. 626)

Podemos dizer que os códigos de programação se constituem em signos que representam ações, assim, esses códigos formam uma língua específica que demanda algum conhecimento técnico para manipulá-la e, como os alunos ainda tinham pouco contato com ela, ter o domínio diretamente em uma folha de papel, onde não é possível ter um *feedback* imediato daquilo que se fez, trouxe a dúvida sobre esse signo e acabou ocasionando em uma grande quantidade de erros. Essa pouca familiaridade com os códigos, quando se trabalha em uma aula plugada, fica mitigada na medida em que os alunos, por tentativa e erro, vão observando os resultados na tela e avançando.

Esse fato foi destacado pelos próprios alunos no momento da entrevista. Conforme já apresentamos nas descrições das aulas, quando os alunos foram instigados a fazerem uma comparação entre as atividades plugadas e desplugadas, a maior parte dos alunos firmou que trabalhar no computador é mais simples devido a possibilidade de avançar espontaneamente a partir do *feedback* imediato do computador.

Os alunos do 7C afirmaram que, no computador, "*pode ir fazendo, fazendo, tentando*" *(SIC),* concluindo que avançam por "*tentativa e erro mesmo*" *(SIC)*. Já uma aluna do 7A diz que *"lá no computador a gente mandava fazer alguma coisa, e tipo... já dava pra ver mais ou menos como ficava. Aí aqui não, tipo... ela [*aponta para a colega*] podia entender de outra forma [*aluna gesticula com os braços representando uma confusão*], lá no computador era só aqueles comandos mesmo" (SIC).* Uma aluna do 7B afirma que no computador "*dava pra você ver, aí se você fizesse errado você podia tirar e fazer diferente*" *(SIC)*.

Todas essas falas ocorridas durante a entrevista das três turmas corroboram nossa afirmação de que o *feedback* imediato proporcionado pelas tecnologias digitais é algo único,

que abre a possibilidade de um trabalho, não melhor, mas diferente daquele feito de forma desplugada, no papel. Essa possibilidade de tentativa e erro novamente nos remonta à espiral de aprendizagem discutida por Valente (2005a) que, dificilmente aconteceria em um trabalho exclusivamente desplugado.

A parte final dessa aula consistia em ajudar a "lagarta comilona" se alimentar sem ser pega pelo pássaro predador. Na primeira aplicação dessa atividade (turmas 7B e C) os alunos utilizaram setas para construir *algoritmo*. Observamos ainda muita dificuldade na utilização das setas, especialmente pela atividade ser mais longa e, assim, para a segunda aplicação (7A) ajustamos passando a utilizar códigos do Scratch e não mais as setas. Essa modificação no formato da linguagem para representar o *algoritmo* permitiu que os alunos avançassem mais nas soluções, possivelmente por terem tido contato e explorado essa linguagem na aula anterior que aconteceu de forma plugada.

Nessa aula os alunos puderam explorar especialmente os conceitos computacionais de *sequências* e *loops* na forma desplugada, porém, utilizando os mesmos blocos de programação da aula plugada. A exploração desses conceitos computacionais em formatos distintos (plugado e desplugado), com grau de complexidade crescente, nos dá uma pista de uma possível potencialização do PC. O grande número de erros na interpretação do comando de giro nos leva a crer que seria necessário mais tempo de exploração desse comando.

5.5 Aulas 5, 6 e 7

Optamos por trazer a discussão dessas três aulas em uma só seção, visto que todas elas compõem o desenvolvimento do projeto Scratch. O objetivo desse projeto foi explorar os conceitos trabalhados nas aulas anteriores, de forma plugada e desplugada, na resolução de um problema por meio do referido software.

Como o tempo para o desenvolvimento desse trabalho era curto, optamos por levar o problema já pronto para os alunos, mesmo tendo a consciência que de tal modo alguns poderiam não ter o mesmo envolvimento do que se eles próprios tivessem determinado o que desenvolver com Scratch.

Na primeira aplicação da oficina, logo na quinta aula (a primeira com o Scratch), já iniciamos problematizando a questão da COVID-19 que foi a temática central dos problemas que seriam resolvidos com Scratch, e então entregamos às equipes as folhas para resolução. O mesmo problema de utilização dos códigos que discutimos na seção anterior aconteceu. Os

alunos não sabiam por onde começar devido à pouca familiaridade com Scratch e, assim, para as outras turmas, iniciamos primeiro desenvolvendo um programa com os alunos buscando familiarizá-los com Scratch para só então entregar a eles o problema relacionado à COVID-19.

Esse pequeno ajuste no formato de trabalho trouxe uma melhora considerável no desenvolvimento dos alunos. Foi possível perceber muito mais empenho e desenvoltura quando eles já estavam melhor direcionados ao o que fazer e tinham alguma noção de como fazer.

Quando elaboramos os problemas para serem resolvidos por meio do Scratch, nossa expectativa era de que os alunos trabalhassem a *resolução do problema* de forma mais autônoma, buscando meios para avançarem em seus trabalhos, seja com auxílio dos colegas, seja fazendo buscas na internet. Essa expectativa estava intimamente relacionada ao que Papert (1994) chamou de matética, que seria a capacidade de construir conhecimentos sem serem ensinados, ou seja, por meio de explorações os alunos seriam capazes de desenvolver aprendizagens.

A experiência inicial que tivemos com a turma 7B, que foi a primeira a iniciar o projeto Scratch, nos mostrou que, problematizar a questão da COVID e logo em seguida entregar o problema para iniciarem a resolução de forma independente, sem terem nenhum conhecimento prévio recente sobre o Scratch, é algo falho e que acaba por desanimar os alunos. A nossa expectativa de que eles fossem buscando meios para avançar no trabalho de forma independente foi frustrada, visto que os alunos ficaram sem saber como fazer o trabalho e, por isso, logo se dispersaram e a aula ficou muito "conversada", não sendo possível que eles avançassem conforme gostaríamos. Por esse motivo, na segunda aula dessa turma, iniciamos resolvendo um pequeno problema no Scratch juntamente com eles (criar um programa para calcular a média aritmética entre dois valores) para só então eles retomarem a questão da COVID. Com esse pequeno programinha introdutório sendo resolvido junto com os alunos observamos um considerável avanço no trabalho.

A partir dessa experiência, para a turma 7C, que foi a segunda a trabalhar o problema COVID, já iniciamos com o programa de calcular a média aritmética entre dois números, para só então entregar os problemas COVID para as equipes. Nessa turma já foi possível perceber que essa pequena introdução é essencial para que os alunos possam se envolver mais com o trabalho e explorarem o problema de forma mais independente.

Essa atitude de iniciar a aula desenvolvendo um programa juntamente com os alunos pode parecer contrária aos princípios do construcionismo, especialmente porque estávamos dando instruções aos alunos de como proceder e isso está mais relacionado ao instrucionismo, que seria uma filosofia oposta ao construcionismo (PAPERT, 1994). Porém, nossa experiência

prévia nos mostrou que os alunos não conseguiram buscar meios para solucionar o problema, seja pela inexperiência com o Scratch, seja pela idade dos participantes da oficina, seja pelo formato de ensino tradicional que estão acostumados no dia a dia, no qual recebem instruções do que e como fazer.

Esse tipo de atividade em um formato mais "independente" dos alunos perpassa por uma inovação pedagógica mais profunda, na qual deveria haver uma redefinição do papel do professor e uma passagem do aluno da periferia para o centro do processo de ação e construção de conhecimento (FINO, 2016). Entendemos que nossa atitude de introduzir o Scratch resolvendo uma atividade juntamente com os alunos não invalidou o caráter construcionista proposto para nossa atividade, mas serviu para nos mostrar que é necessário que os alunos tenham algum conhecimento prévio para que sejam capazes de explorar e avançar por conta própria naquilo que foi proposto.

A seguir transcrevemos um diálogo entre pesquisador e uma equipe sobre o problema da COVID que estava sendo resolvido via Scratch.

Pesquisador: E aí, carinha. Como é que tá o projeto? (SIC)
Aluno 1: Tô fazendo ele pôr o alerta(SIC)
Pesquisador: Oi? (SIC)
Aluno 1: Tô fazendo ele pôr o alerta(SIC)
Pesquisador: Beleza! Ah, já achou o comando "se", isso mesmo... Agora "se" o quê que você tem que fazer, meu amigo? (SIC)
Aluno 1: Se "leitos" dividido por "pacientes" é.... se "leitos" dividido por "pacientes" for alguma coisa, tenho que dar o alerta(SIC)
Pesquisador: Isso... Que alguma coisa é essa? Pega a folhinha lá pra gente ver. Empresta a folhinha pra nós aqui, meu amigo (pesquisador pede a folha a outro aluno da equipe). *Dá uma olhada aqui, óh.* (pesquisador mostra trecho do enunciado na folha para o aluno e faz a leitura). *Quando o valor estiver muito baixo, inferior a 3 dias, por exemplo, deverá emitir um alerta. Então, se "leitos" dividido por "pacientes" for menor que... (SIC)*
Aluno 1: que 3(SIC)
Pesquisador: Pode crê! (SIC)

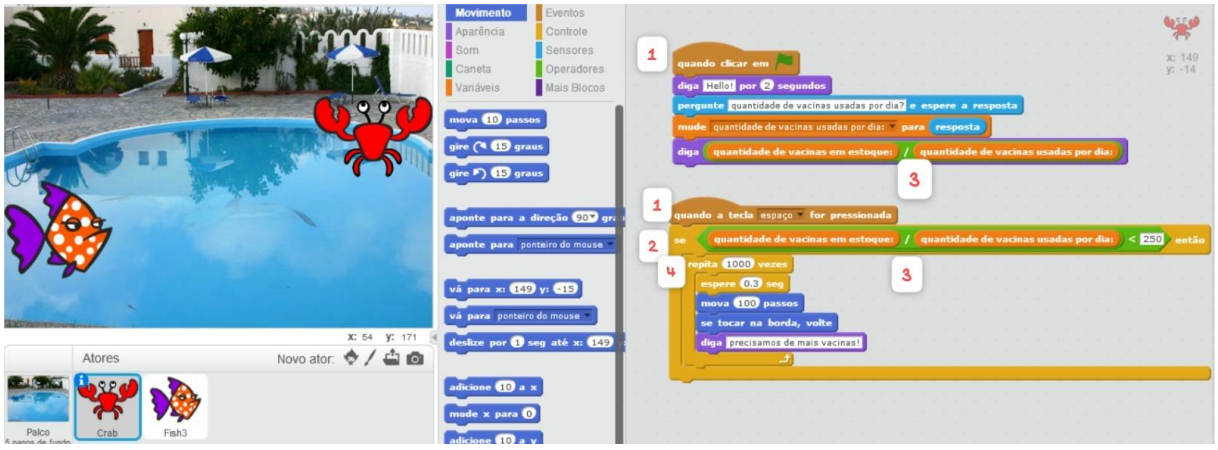
Pelo diálogo, é possível observar que a equipe, após a introdução prévia ao Scratch pelo pesquisador por meio do programa inicial de cálculo de média, já estava avançando para uma questão *condicional* e tinha encontrado o comando "se" para utilizar, porém, apresentou dúvida

em sua utilização. O pesquisador auxilia os alunos em como utilizar o código e o diálogo finaliza dando a entender que os alunos compreenderam como inserir o código em seu programa.

Vale ressaltar que o programa desenvolvido junto com os alunos objetivando introduzir o Scratch não fazia nenhuma menção à utilização do comando de *condicional* e, no programa feito por eles da COVID, era necessária essa utilização. Conforme ilustrado pelo diálogo transcrito acima, os alunos conseguiram avançar de forma independente até encontrarem esse comando. Tal fato pode indicar que os alunos não desanimaram ao se depararem com esse desafio, e buscaram de forma independente esse novo comando para solucionarem essa parte do problema. Isso nos permite novamente inferir que a introdução prévia ao *software* de programação não minimizou o caráter construcionista da atividade, mas potencializou, visto que os alunos já tinham algum conhecimento sobre o *software* e isso os permitiu avançar de forma independente.

No contexto dos programas desenvolvidos pelos alunos, vários outros conceitos computacionais (BRENNAM; RESNICK, 2012) foram explorados de forma independente pelos alunos. A Figura *62* apresenta alguns destaques nos códigos de programação referente a um personagem do programa desenvolvido por uma das duplas na turma 7A.

Figura 62 - Programa Scratch – personagem 1



Fonte: o autor

Conforme é possível observar na imagem:

- o item 1 destaca o conceito de *evento* (quando clicar na bandeira verde ou quando pressionar o espaço o bloco de comandos abaixo é executado);

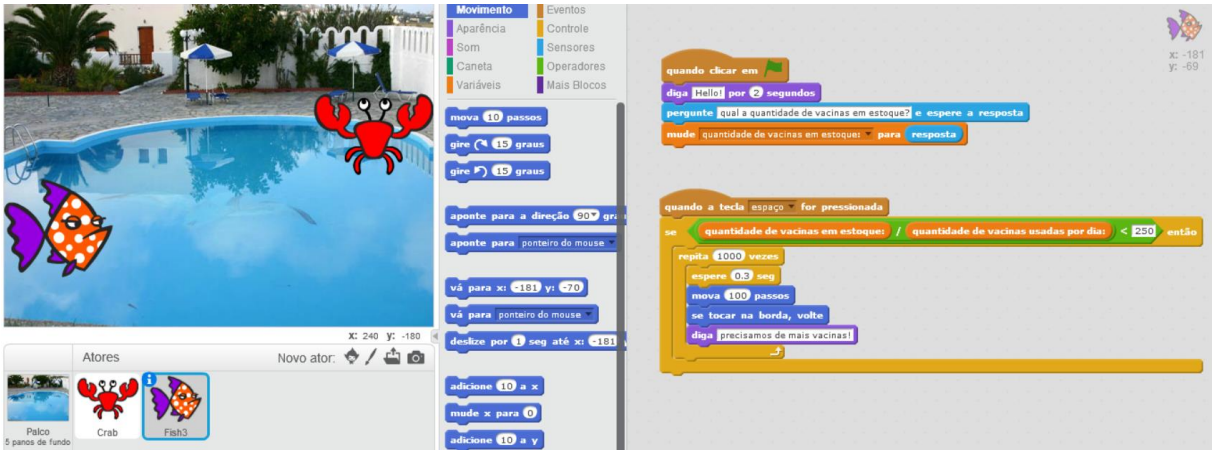
- o item 2 destaca o conceito de *condicional* (é feita uma verificação entre valores de variáveis e o bloco seguinte é executado quando essa condição for satisfeita);

- o item 3 destaca os conceitos de *operadores* e *dados* (operador de divisão e de comparação – blocos verdes e os dados são as variáveis “quantidade de vacinas em estoque” e “quantidade de vacinas usadas por dia” que armazenam os dados inseridos pelo usuário do programa);

- o item 4 destaca o conceito de *loop* (há uma repetição de 1000 vezes os comandos internos ao loop que formam uma movimentação na tela do personagem).

Esses códigos exibidos na Figura *62* são referentes ao personagem “caranguejo”, porém, há outros comandos construídos para o personagem “peixe”, conforme está ilustrado na Figura *63*.

Figura 63 - Programa Scratch – personagem 2



Fonte: o autor

É possível observar que o conceito de *paralelismo* também foi explorado na medida em que há comandos que são executados em paralelo, ou seja, quando a bandeira verde é clicada há execução ao mesmo tempo de códigos para os dois personagens.

É importante destacar como a ludicidade do Scratch chama atenção dos alunos. Conforme relatamos, grande parte dos alunos começou seu projeto se preocupando com a aparência do programa, modificando personagens e plano de fundo, procurando formas de fazer movimentos etc. Como o tempo para a construção dos projetos era limitado, o pesquisador instigou os alunos várias vezes para que avançassem na resolução da proposta e procurassem deixar as questões visuais para o final. Essa preocupação em relação ao tempo dispendido para o desenvolvimento de um projeto também é apresentada por Brennam e Resnick (2012) ao solicitarem aos programadores que remixassem um projeto Scratch. Por quanto tempo eles deveriam trabalhar nessa tarefa?

A limitação de tempo para esse tipo de trabalho é um fator que merece atenção ao se pensar em atividades com caráter mais exploratório, ainda mais quando se trabalha com um *software* como Scratch, que permite uma grande variedade de explorações por parte dos alunos. Mesmo com o tempo limitado, foi possível que os alunos explorassem uma grande quantidade de conceitos computacionais ao longo desse trabalho, conhecessem recursos novos que ainda não haviam sido explorados nas aulas anteriores e utilizassem os pilares do PC para a construção do projeto Scratch, nos dando assim mais uma pista de uma possível potencialização do PC nesses alunos.

5.6 Casos que emergiram da análise

Ao procedermos a análise dos dados, algumas questões nos chamaram a atenção e abriram possibilidades para investigarmos mais à fundo dois casos que apresentamos na sequência.

5.6.1 O caso Joaquim[15]

Quando aplicamos a primeira atividade desplugada em sala, na qual os alunos deveriam pensar em alguma atividade cotidiana e descrevê-la em um formato de passo a passo (*algoritmo*) como se estivessem ensinando a um robô a resolver aquela tarefa, tínhamos o objetivo de mostrar aos alunos que nossas tarefas cotidianas podem ser entendidas como um problema a ser resolvido e que podemos fazê-lo de forma estruturada, seguindo uma sequência de passos, utilizando o Pensamento Computacional. A maior parte dos alunos procedeu conforme era esperado nessa atividade, porém, Joaquim fez algo que nos chamou a atenção.

Na Figura *49*, apresentamos sua solução para essa atividade na qual ele afirma que gostaria apenas de um robô companheiro, que o ouvisse. Ao deparar com essa situação, o pesquisador o questiona e solicita que ele complemente sua solução e então, na mesma folha, descrevesse o que gostaria que o robô fizesse. Ele então apresenta no formato de tópicos que o robô deveria conversar com ele e jogar vídeo game, conforme é possível observar nessa mesma figura.

Alguns dias depois de finalizada a oficina na escola, ao iniciar a análise dos dados e deparar com essa situação, o pesquisador entrou em contato com a professora de matemática

[15] Utilizamos um nome fictício para preservar a identidade do aluno

do ano escolar e buscou mais informações sobre o aluno. Conforme foi relatado, esse aluno sempre afirma não ter amigos, apesar de ter boa convivência em sala com os colegas. Outro fato importante por ela relatado é que ele quase sempre dorme durante as aulas e a família não cuida muito. Inclusive esse aluno já teve alguns acompanhamentos do núcleo de psicologia da escola. A professora afirmou que ele é um pouco depressivo e, nos intervalos, está sempre junto de uma aluna de outra turma que também é assim. Conforme relatado, eles ficam juntos no recreio, porém, sempre muito calados.

A partir disso passamos a analisar as demais atividades desenvolvidas por esse aluno ao longo da oficina para tentar observar como ele procedeu ao longo dos trabalhos.

Na segunda aula a atividade desenvolvida foi desplugada utilizando folhas de papel com quadriculados. Para essas atividades os alunos formaram duplas, sendo um deles o programador e o outro o computador. O aluno programador deveria "programar" o colega computador para guiar um objeto até determinado alvo. Antes de iniciar essa atividade o pesquisador solicitou à classe que formassem duplas e, exatamente Joaquim ficou sem uma dupla, pois naquele dia havia uma quantidade ímpar de alunos em sala.

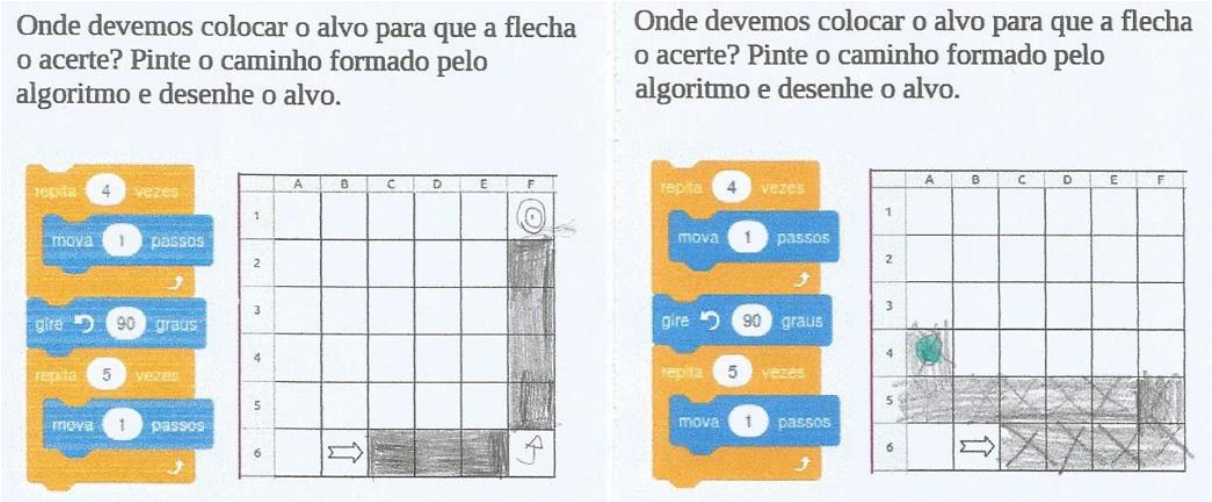
Esse fato corrobora a fala da professora de matemática de que esse aluno dorme muito nas aulas e por isso é um pouco discriminado pelos colegas de classe nas atividades conjuntas, visto que o consideram como aquele aluno que não produz muito.

Nesse formato, infelizmente o Joaquim não acrescentou seu número de chamada na atividade da dupla que ele integrou, impossibilitando o pesquisador de localizar quais foram as resoluções dadas pela dupla que Joaquim estava junto. O mesmo aconteceu para as demais atividades, porém, apesar de não termos conseguido observar as soluções apresentadas por Joaquim ao longo da oficina, conseguimos perceber um ponto positivo que possivelmente foi desencadeado pelo formato de nosso trabalho. Em cada dia de atividades esse aluno esteve junto com colegas diferentes, não se mantendo "fixo" a uma única dupla. Olhando para essa situação sob um viés positivo, inferimos que a oficina permitiu a esse aluno um convívio ao longo desses dias com vários colegas diferentes, o que pode ter contribuído para minimizar esse caráter "solitário" que ele deixou transparecer pela solução de sua primeira atividade ao relatar que gostaria de um robô companheiro.

5.6.2 O caso TDAH

Quando procedemos a análise dos dados da aula 4 da turma 7A, desplugada, uma dupla nos chamou a atenção especialmente na segunda atividade por ter sido a única dupla a desenhar um percurso diferente do esperado. Todas as outras 13 duplas desenharam o percurso como era esperado, mas a dupla em questão resolveu de forma diferente. Na Figura *64* ilustramos, à esquerda, uma resolução conforme era esperado e à direita a resolução feita pela dupla em questão.

Figura 64 - Resolução atividade 2 da aula 4



Fonte: o autor

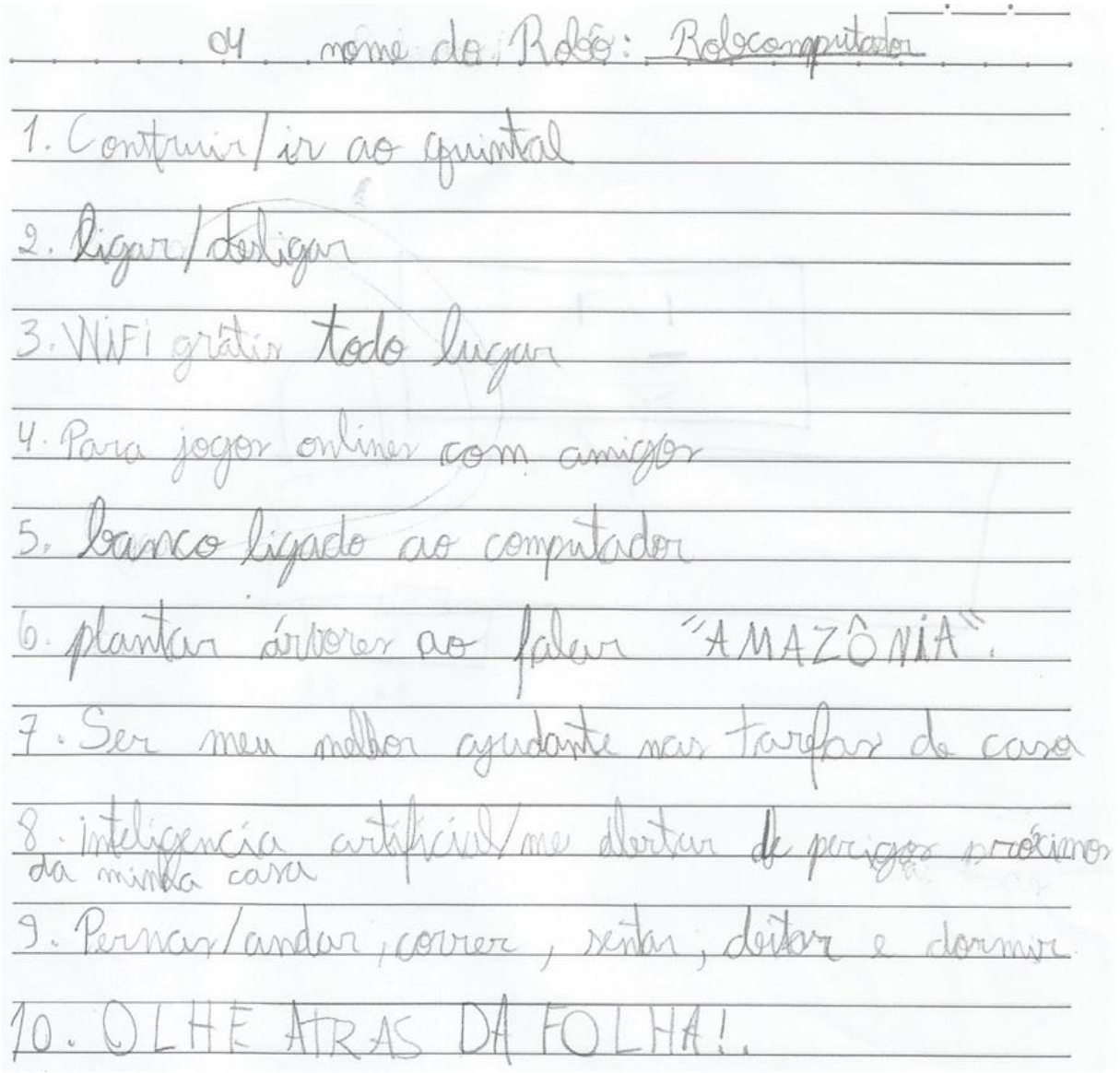
É possível inferir que os alunos se confundiram com a "setinha" que existe na parte inferior do bloco "repita". Dá a entender que eles olharam para essa setinha e a interpretaram como se fosse um bloco de giro e chegaram na resolução por eles construída. Essa confusão pode estar associada ao pouco contato com os blocos de programação do Scratch conforme já discutimos anteriormente, mas o que chamou realmente a atenção foi o fato de essa ter sido a única dupla da turma a proceder dessa forma enquanto todas as outras "acertaram" essa resolução. Na outra atividade desplugada com esse mesmo formato (os alunos recebiam o código e tinham que construir o percurso no quadriculado), essa dupla também fez uma resolução bem distinta daquilo que era esperado.

A partir disso investigamos um pouco mais sobre esses alunos e, em um novo contato com a escola, descobrimos que ambos alunos dessa dupla possuem laudo médico de Transtorno de Déficit de Atenção e Hiperatividade (TDAH). Com essa nova informação retomamos as atividades desses alunos. Na Figura *65* apresentamos a primeira atividade desplugada feita em

sala, na qual eles deveriam pensar em um problema cotidiano e descrever passo a passo a resolução desse problema como se estivessem "ensinando" a um robô a resolver esse problema.

Conforme é possível observar, esse aluno não compreendeu bem o que deveria ser feito mesmo tendo sido atendido individualmente para tirar dúvida de como fazer pelo pesquisador. Em forma de tópicos, ele foi anotando "coisas" que ele gostaria que o robô tivesse ou fizesse. No verso de sua folha fez o desenho de um computador com braços e pernas sorrindo.

Figura 65 - Atividade 1 aluno A

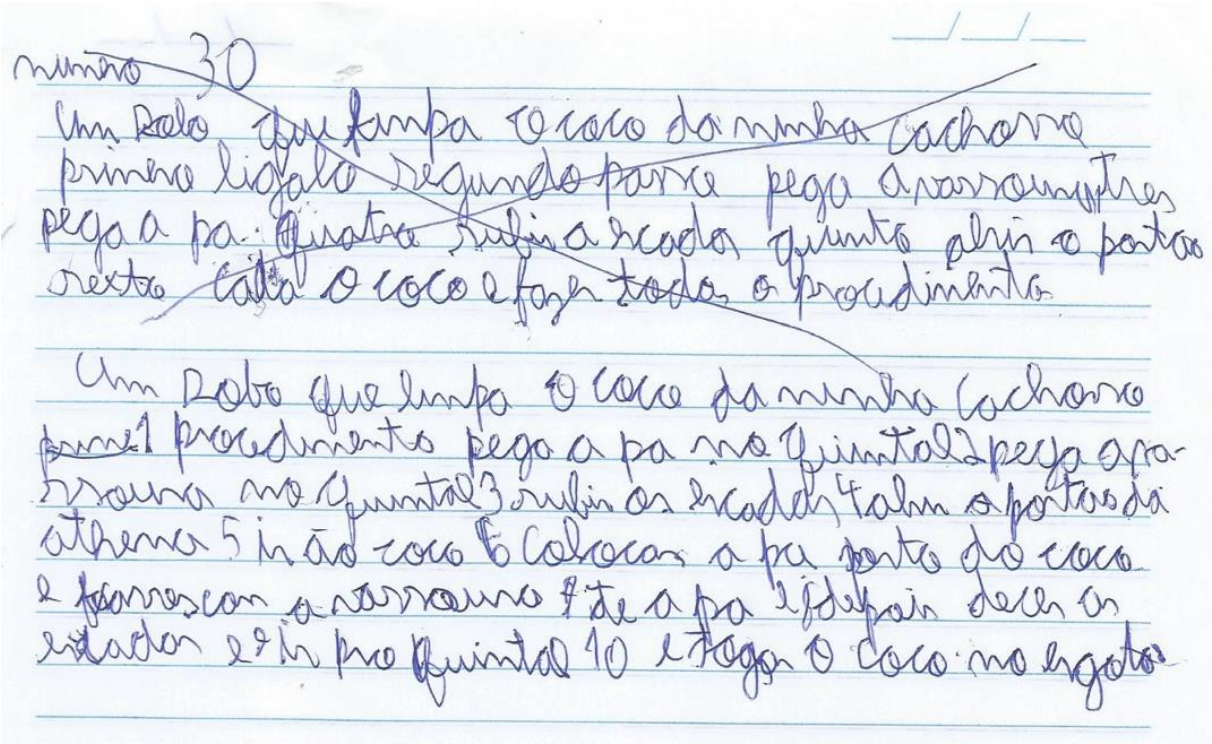


Fonte: o autor

Já o segundo aluno dessa dupla conseguiu pensar em uma tarefa cotidiana (recolher cocô do cão) e tenta descrever essa tarefa na forma de passos, porém, ele descreve na forma de um

parágrafo. Quando ele havia “terminado” a atividade, chamou o pesquisador para mostrar sua solução e então foi orientado a escrever na forma de tópicos, indicando cada passo que o robô deveria seguir para resolver o problema de “recolher o cocô do cão”, mas ainda assim, com a segunda orientação, esse aluno mantém o formato de parágrafo, mas agora numerando cada passo. Sua atividade está ilustrada na Figura *66*.

Figura 66 - Atividade 1 aluno B



Fonte: o autor

Olhando para esse bloco de atividades desplugadas dessa dupla, inferimos que esse formato de trabalho, ao menos para esses alunos, não alcançou o que almejamos, pois eles não conseguiram demonstrar no papel a descrição de uma atividade cotidiana como algoritmo ou ler uma sequência de códigos e transformá-la em um percurso no quadriculado, porém, o mesmo não aconteceu para as atividades plugadas.

Na primeira aula plugada, quando tiveram contato com Scratch, eles conseguiram, com algum auxílio do pesquisador, construir um percurso e programar a flechinha para percorrer esse trajeto. Já no projeto maior que deveriam resolver o problema COVID, conseguiram avançar bastante. Por meio dos vídeos que gravamos da tela do computador, capturamos a imagem ilustrada na Figura *67* do início da aula 7. É possível observar que os alunos já tinham

conseguido utilizar as variáveis juntamente com os comandos de pergunta e operador matemático. Em um determinado momento dessa aula, para toda a classe, o pesquisador relembra da segunda etapa da atividade que eles deveriam construir alguma animação em seu programa, de forma livre. Após esse “lembrete” do pesquisador, os alunos começaram a trabalhar nessa animação e a presente dupla fez diversas modificações em seu programa, ajustando planos de fundo, inserindo novos personagens e construindo diálogos e movimentos. Uma nova captura de tela está ilustrada na Figura *68* com o programa dessa dupla já alterado.

Figura 67 - Momento inicial aula 7



Fonte: o autor

Figura 68 - Momento final aula 7



Fonte: o autor

Um destaque interessante a ser acrescentado é que, para essa turma, conforme já relatamos anteriormente, a apresentação dos trabalhos aconteceu na própria sala de informática, com o pesquisador passando em cada um dos computadores e olhando os trabalhos. Essa dupla, após apresentar seu trabalho, continuou modificando seu projeto, procurando na internet áudio para colocar em seu programa e ajustando falas dos personagens. Esse fato chamou muita atenção visto que o trabalho já havia sido "finalizado" e eles já tinham apresentado, porém, os próprios alunos quiseram continuar seu trabalho mesmo sabendo que já não haveria mais apresentação e nenhuma outra pessoa veria seu trabalho.

Essa dupla de alunos nos fez refletir sobre a importância do professor em "perceber" na sala de aula os alunos com mais dificuldades de concentração e, dentro das possibilidades, diversificar os meios de abordagem dos diversos conteúdos, visto que para as atividades desplugadas essa dupla não se saiu bem, muito diferente do que ocorreu para a parte plugada da oficina.

Caso a oficina de Pensamento Computacional tivesse sido oferecida apenas no formato desplugado, com todas as atividades no formato que fizemos, teríamos concluído que esses alunos não se envolveram no trabalho, ou não tinham entendido quase nada sobre o PC. Na verdade, esse formato de trabalho (em sala de aula, demandando leitura e concentração) é um formato que exige uma atenção que esses alunos, conforme laudo médico apresentado na

escola, não têm. Porém, no formato plugado, as atividades desenvolvidas no Scratch, permitiram a esses alunos avançarem mais em seus trabalhos.

5.7 Uma síntese em relação a análise

Com objetivo de sintetizar nossa discussão, procuramos elaborar tópicos em um quadro que apresenta, aula a aula, qual era a expectativa para aquela aula, ou seja, o objetivo do professor e o que os dados apresentaram.

Quadro 3 - Resumo das discussões

	O que foi planejado? (expectativa/perspectiva do professor)	**O que os dados apresentaram? (realidade/perspectiva do aluno)**
Aula 1 (desplugada)	- Noção de PC; - Relacionar o PC ao cotidiano; - O que é um algoritmo; - Descrever uma tarefa do cotidiano na forma de algoritmo.	- *Resolução de problemas* cotidianos no formato de *algoritmo*; - Atividades que não condiziam com o esperado e que, provavelmente, foram influenciadas por fatores externos (Souza et al. (2020))
Aula 2 (desplugada)	- Estabelecer uma comunicação entre programador e computador; - Construir *algoritmo* utilizando comandos padronizados (setas) para representar percurso;	- Dificuldade na comunicação entre os alunos, especialmente em relação a lateralidade; - *Algoritmos* representados por setas; - *Algoritmos* errados com percurso correto: cola possivelmente influenciada pelo paradigma de ensino tradicional (Leão (1999))
Aula 3 (plugada)	- Contato inicial com programação em blocos; - Construção de *algoritmos* utilizando linguagem de programação em blocos com Blockly Games e Scratch; - Conhecendo estruturas de repetição e condição;	- Diversos conceitos computacionais (Brennam e Resnick (2012)) explorados pela programação em blocos; - Ressignificação do erro proporcionado pelo *feedback* imediato e ocorrência da espiral de aprendizagem proposta por Valente (2005a)
Aula 4 (desplugada)	- Construção e interpretação de *algoritmos* mais	- Construção de *algoritmos* com setas;

	complexos utilizando mesma linguagem de programação em blocos; - Estruturas de repetição e comandos de giro; - Resolução de problemas a partir da programação;	- Dificuldade com os comandos de giro devido a nova linguagem; - *Algoritmos* errados com percurso correto: cola possivelmente influenciada pelo paradigma de ensino tradicional (Leão (1999))
Aulas 5, 6 e 7 (plugada)	- Resolver problemas relacionados à COVID-19 por meio do desenvolvimento de projeto Scratch:	- Pilares do PC emergiram do desenvolvimento da maioria dos projetos; - Conceitos computacionais explorados espontaneamente pelos alunos;
Aula 8 (finalização)	- Discussão dos projetos; - Entrevista coletiva.	- Mesmo sem o projeto funcionar, algumas equipes fizeram questão de mostrar o que haviam feito e explicar o que haviam planejado para terminar o trabalho; - Uma dupla não quis ir até a frente da turma apresentar seu trabalho; - No geral, avaliação da oficina foi positiva na visão dos alunos, sendo sugerido "mais aulas com o computador"

Fonte: o autor

Nesse quadro procuramos apresentar de forma sintética os principais tópicos que surgiram ao longo das análises. Conforme pudemos observar, conceitos computacionais e os pilares do PC foram tangenciados ao longo de toda a oficina, de forma gradativa e evolutiva em relação ao grau de complexidade, abrindo oportunidade para potencialização do PC dos alunos envolvidos nesse trabalho.

Além dos tópicos já apresentados no quadro, procuramos apresentar abaixo, aula a aula, uma pequena síntese de nossos "achados" que a aplicação da oficina, bem como sua análise, nos permitiu observar.

Aula 1: o termo potencializar se mostrou condizente ao percebermos a existência de um estilo de PC inerente aos alunos, assim, ao dizermos que a oficina potencializou esse estilo de pensamento, afirmamos que já existia nesses alunos um estilo de pensamento estruturado que foi potencializado pela oficina.

Aula 2: a potencialização do PC ocorrida nessa aula, especialmente no desenvolvimento dos algoritmos com as setas, dependeu, em muitos casos (como no diálogo ilustrado na análise) da mediação do professor, reforçando o importante papel de auxiliar os alunos na construção de seu conhecimento, evitando que desanimem das atividades ao encontrarem obstáculos mais complexos.

Aula 3: o erro, no contexto das atividades plugadas, é ressignificado pelo aluno na medida em que se torna um obstáculo a ser ultrapassado. Com isso, a espiral de aprendizagem (VALENTE, 2005a) acontece e, à medida em que os pilares do PC e os conceitos computacionais vão sendo explorados na resolução dos problemas, os alunos vão tendo seu PC potencializado, especialmente porque nesse contexto, na maioria das vezes, o crivo sobre a resolução do problema (o programa Scratch) parte do próprio aluno, visto que ele mesmo observa seu resultado e reformula seu programa até alcançar o resultado desejado.

Aula 4: os conceitos computacionais, bem como os pilares do PC foram explorados no formato desplugado de forma mais complexa, permitindo trabalhar novos conceitos e comandos. As dificuldades ocorridas nessa aula reforçam que o *feedback* imediato das tecnologias digitais permitiria aos alunos avançarem de forma mais independente durante as atividades, potencializando seu PC por meio da espiral de aprendizagem.

Aulas 5, 6 e 7: os alunos puderam explorar todos os conceitos computacionais bem como os pilares do PC, construindo animações e histórias para resolverem os problemas pelo Scratch e, com isso, potencializaram seu PC. Destacamos a importância de terem algum conhecimento prévio em relação ao Scratch para conseguirem avançar nos trabalhos (pelo menos para a idade dos envolvidos em nossa intervenção).

Além dos tópicos apresentados para cada aula, destacamos ainda os casos Joaquim e TDAH que os dados nos apresentaram. O formato híbrido da oficina que propusemos permitiu a inclusão desses alunos e, assim, foram capazes de se integrarem e conseguirem produzir ao longo das aulas.

Para além disso, o formato híbrido da oficina também nos permitiu observar a assimetria existente entre professor e aluno no contexto da sala de aula tradicional, o que não aconteceu nas aulas plugadas, dentro da sala de informática. Na sala tradicional, o professor é o detentor do conhecimento e os alunos o veem como referência, questionam se a atividade está correta o tempo todo para ele; no contexto da sala de informática (pelo menos em nossa oficina), o professor debatia com os alunos as soluções em uma relação de horizontalidade, apresentando maior proximidade com os alunos.

Finalizamos nossa análise ressaltando que, ao desenvolvermos a oficina de PC objetivando potencializar esse estilo de pensamento nos alunos, estamos contribuindo para que se tornem construtores de tecnologia. Para nós, desenvolver essa capacidade de construção de tecnologias possui grande relevância, e observamos tal fato sob duas perspectivas.

A primeira, para nós, está relacionada à capacidade de construir tecnologias para melhorar sua realidade, sua vida. No geral, as tecnologias, digitais ou não, estão em nossa vida com objetivo de melhorar, facilitar nosso dia a dia. Ao potencializarmos o PC no sentido de sermos capazes de construir tecnologias, solucionar problemas com auxílio do computador, avançar em um estilo de pensamento estruturado, desenvolvemos a capacidade de construirmos soluções para problemas de nosso cotidiano. Por exemplo, criar uma planilha eletrônica ou um programa para resolver problemas relacionados ao meu trabalho. Com isso você não depende de usar algum recurso pronto, adaptado, mas você mesmo tem a capacidade para desenvolver sua solução.

Ou ainda, caso exista alguma solução parecida com o que você precisa, você será capaz de ajustar aquela solução para sua realidade. Por exemplo, é possível pegar um projeto Scratch no banco de dados do site e modificar o código (alterar personagem, acrescentar funções ao programa etc). A isso Resnick (2012) chama de remixar.

Outra perspectiva é em relação a criticidade. Avançar no desenvolvimento do PC, conhecer um pouco das formas de representar algoritmos por meio de linguagens computacionais, permite ter uma visão do que está por detrás dos programas, sites, aplicativos. Entender um pouco de como são criados abre oportunidade para compreender que, às vezes, podem ter objetivos maliciosos, como por exemplo capturar dados por meio de phishing. Assim, ao ter o PC potencializado, é possível ter um olhar mais crítico em relação às tecnologias.

6 CONSIDERAÇÕES FINAIS

Quando nos propusemos a desenvolver uma pesquisa objetivando compreender como o Pensamento Computacional pode ser desenvolvido por meio de uma oficina de Pensamento Computacional, muitos desafios se puseram até que conseguíssemos chegar aqui. A pandemia de COVID-19 dificultou as atividades presenciais e, por esse motivo, acabamos fazendo menos intervenções do que gostaríamos.

No contexto da aplicação, fatores como troca de sala pelos alunos e organização das turmas atrasaram o início das aulas, o que acarretou na redução do tempo, deixando alguns momentos mais corridos. Apesar desses pequenos contratempos, as intervenções se mostraram frutíferas.

Para alcançar o primeiro objetivo específico de nossa pesquisa, que era analisar como cada atividade, plugada e desplugada, pode contribuir, ou não, para a potencialização do PC, observamos cada uma das atividades, analisando as produções e manifestações dos alunos ao longo de todo o trabalho.

Logo na primeira atividade, buscamos observar como os alunos analisavam e resolviam problemas do cotidiano por meio de uma sequência de passos. Com essa atividade, o objetivo da oficina de explorar conceitos computacionais foi alcançado no momento em que os alunos, de forma desplugada, descreveram no papel tarefas de seu cotidiano em uma *sequência* de passos, construindo uma espécie de *algoritmo* para solucionar aquele problema. Ao lançarmos nosso olhar para essa primeira atividade da oficina, observamos como é aplicado o PC "rotineiro", "cotidiano" a situações do nosso dia a dia. Os alunos construíram uma *sequência* de passos e o *algoritmo* para resolver um problema de seu cotidiano por meio de uma linguagem natural. Como não houve nenhuma especificação, a descrição passo a passo se deu utilizando frases em português que resumiam cada passo a ser dado para alcançar a solução daquela tarefa.

Tal fato reforça a questão que dissemos sobre o PC ser potencializado, ou seja, adaptado. Conforme a atividade revelou, temos a ideia do que seja o PC e o utilizamos em nosso dia a dia, porém, precisamos fazer "adaptações" para aplicar esse pensamento nas mais diversas tarefas. É preciso compreender que, quando descrevemos um algoritmo computacional, por exemplo, utilizamos uma linguagem de programação específica, códigos ou blocos para programar. A primeira atividade da oficina permitiu observarmos que realmente existe um pensamento estruturado e sequencial naqueles alunos que pode ser adaptado, aperfeiçoado, logo, potencializado.

Essa atividade, de alguma maneira, oportunizou aos alunos explorarem também o pilar do PC relacionado a *decomposição* de problemas. Mesmo que de forma transparente, eles puderam explorar, logo na primeira atividade desplugada, que para programar um robô para "dar banho na minha cachorra" (atividade desplugada apresentada por um aluno), não bastava dizer ao robô "dê banho na cachorra", mas era preciso *decompor* esse problema em outros menores, mais simples de gerenciar, em uma *sequência* de passos formando um *algoritmo*.

Assim, conforme descrito por esse aluno, primeiro o robô deveria "pegar sabonete, shampoo e toalha", depois "ir até o quintal", "pegar a mangueira", "prender o portão para o cachorro não fugir", "ligar a mangueira" etc. É possível perceber ainda que o pilar *abstração* do PC também se fez presente nessa atividade na medida em que eles, para descreverem passo a passo a resolução da questão, ignoraram questões irrelevantes à solução do problema, como por exemplo, a cor da mangueira. Esse fato foi deixado de lado ao dar o comando e, por isso, o "pegar a mangueira" *abstrai* a questão da mangueira ser preta, azul ou amarela. O mesmo pode-se imaginar para a toalha que poderia ter tamanhos e cores diferentes, e assim por diante.

Essa primeira atividade nos permitiu observar que, os alunos ao refletirem sobre suas tarefas cotidianas, procurando *decompô-las* em uma *sequência* de passos formando um *algoritmo,* puderam, em alguma medida, compreender que, para solucionar um problema, um bom caminho é dividi-lo em partes menores, descrevendo cada uma das partes de forma sucinta, ignorando detalhes irrelevantes à solução daquele problema.

Conforme as atividades foram avançando, esses alunos puderam ir percebendo que cada parte desse *algoritmo* deveria estar em uma linguagem que o computador/robô pudesse entender. Tal fato foi melhor explorado na atividade seguinte, na qual os alunos deveriam estabelecer um padrão de comunicação para dizer os movimentos de um objeto no quadriculado até alcançar o objetivo. Assim, o *algoritmo* não era mais descrito com frases, mas simplesmente com setas e, mais adiante, essas setas se tornaram blocos de programação no Scratch que movimentavam personagens, produzia falas e efetuavam cálculos.

Ao refletirem sobre as tarefas/problemas de seu próprio cotidiano e transformarem essas tarefas em comandos a serem ensinados a um robô, os alunos utilizaram conceitos computacionais e aplicaram pilares do PC, o que nos dá uma pista de que, possivelmente, essas atividades podem ter contribuído para a potencialização de seu Pensamento Computacional a partir da *decomposição* do problema e formação de um *algoritmo*.

O objetivo da oficina de proporcionar experiências plugadas e desplugadas que explore conceitos do PC foi sendo tangenciado por todas as atividades de forma gradual e evolutiva. As experiências desplugadas permitiram explorar especialmente os conceitos computacionais de

sequências e *loops* (BRENANN; RESNICK, 2012), além dos pilares *decomposição* de problemas, *abstração* e *algoritmos* (LIUKAS, 2015; BBC BITSIZE, 2015; BRACKMANN, 2017). Para além dos conceitos relacionados diretamente ao PC, foi explorado também o estabelecimento de uma nova linguagem de comunicação entre os alunos, questões relacionadas a lateralidade, além do trabalho em conjunto.

Já as atividades plugadas também exploraram os conceitos de *sequências* e *loops*, além de *eventos*, *paralelismo*, *condicionais*, *operadores* e *dados* conforme foi destacado na Figura 62 e na Figura 63. As imagens ilustram o projeto Scratch sobre a COVID-19, mas esses mesmos conceitos foram explorados também por meio dos jogos Blockly Games, o que também permitiu alcançar o objetivo de explorar os conceitos computacionais por meio da programação em blocos.

Os alunos foram trabalhando cada uma dessas atividades e se envolvendo em suas soluções, avançando gradativamente na complexidade dos conceitos computacionais e explorando-os em um formato híbrido. Tal fato os permitiu construir uma visão mais ampla em relação a esses conceitos computacionais, visto que puderam explorá-los tanto no papel quanto na tela do computador.

O objetivo específico de compreender de que modo a articulação entre atividades plugadas e desplugadas podem favorecer a potencialização do PC, foi alcançado na medida em que observamos que, o avanço gradual de dificuldade entre as atividades plugadas e desplugadas permitiu aos alunos uma visão mais ampla e diversificada em relação ao PC, assim, tal fato nos dá uma nova pista de uma possível potencialização do PC.

Quando elaboramos a oficina com o objetivo de analisar como o Pensamento Computacional pode ser desenvolvido/potencializado a partir de uma abordagem híbrida, não imaginávamos quantas outras coisas poderiam surgir a partir desse trabalho. Os dados mostraram como fatores externos operam fortemente no contexto educacional.

O “caso Joaquim” ilustrou como questões de cunho psicológico estavam incomodando aquele aluno a ponto de ele utilizar a atividade para, de alguma forma, externar seu sentimento. Isso nos leva a refletir sobre quantos “Joaquins” estão espalhados em nossas salas de aula no dia a dia, reforçando a necessidade de termos um olhar ao nosso aluno para além de “mais um aluno”, visto que ele traz para sala de aula consigo toda uma vivência, questões pessoais, projetos de vida, ideais. Isso nos permite inferir que um aluno desatento na aula, desinteressado ou que esteja conversando, pode significar muito mais do que “um aluno que não quer nada”, mas pode ser um aluno que esteja passando por problemas fora da escola, seja familiar ou financeiro. Conversar com esse aluno e entender suas questões pode favorecer seu aprendizado.

Conforme apontamos anteriormente, não pudemos acompanhar mais de perto as atividades desenvolvidas pelo Joaquim, visto que ele não se fixou em uma dupla, fazendo os trabalhos com vários outros alunos. Observamos esse fato sob um viés positivo visto que, pelo formato da oficina, suas questões pessoais foram mitigadas e ele se envolveu com outros colegas, participando dos trabalhos. Mesmo que não tenhamos conseguido analisar especificamente seus trabalhos, em alguma medida, a oficina permitiu que ele interagisse com os demais durante as atividades.

Já o "caso TDAH" nos mostrou como é importante a questão da diversificação da abordagem do professor. A observação desse caso nos permitiu alcançar o objetivo da pesquisa de estudar como o formato híbrido da oficina pode contribuir para repensar práticas educacionais. À primeira vista, aqueles alunos não haviam se interessado pelas atividades desplugadas de PC e, por isso, conversavam muito. Mas as atividades plugadas mostraram que eles tinham interesse pela programação, pela construção de histórias e animações. Tal fato reforça a importância de ter uma abordagem múltipla em sala de aula, utilizando diferentes recursos e formas de trabalho, visto que, conforme já destacamos, em uma sala de aula há uma diversidade enorme de alunos com muitas particularidades. Um único tipo de abordagem dificilmente atenderá a todos.

Assim, o formato híbrido da oficina nos permitiu compreender como é importante que uma prática educacional procure ser o mais abrangente possível, para que possa atender a grande diversidade que existe no contexto de uma sala de aula.

Tanto caso Joaquim quanto caso TDAH nos permitiu alcançar o objetivo de determinar que outros elementos, para além do PC, o formato híbrido da oficina pode fomentar. O formato proposto, além do caráter incremental e misto de abordagem para potencializar o PC, permitiu a inclusão de alunos que, talvez, em um formato tradicional de ensino, não teriam participado e produzido conforme os dados revelaram.

Os dados nos mostraram ainda como o ensino tradicional está arraigado na cultura escolar. Mesmo reforçando com os alunos de que a participação na oficina "não valia ponto" e que era voluntária, eles, o tempo todo, se preocupavam em fazer de forma correta as atividades, chegando a "colar" para entregar ao professor conforme ele gostaria de ver. Tal fato reforça como a cultura de provas e avaliações influencia fortemente na sala de aula, mesmo quando a proposta procura se distanciar do formato tradicionalmente utilizado.

Para além desses fatores, os dados nos revelaram ainda uma importante contribuição do formato híbrido da oficina. Foi possível perceber claramente que, ao desenvolvermos as atividades desplugadas (que ocorreram dentro da sala de aula tradicional dos alunos), os alunos

tinham um comportamento rotineiro do seu dia a dia. Nesse contexto, foi possível observar haver uma espécie de jogo psicológico, no qual o professor é a autoridade nesse meio. Os dados revelaram os alunos o tempo todo chamando o professor para mostrar sua atividade e pergunta-lo se estava correto, além do fato de eles "colarem" os labirintos para entregarem a atividade de forma "correta" para o professor. Tal fato reforça a assimetria entre professor e aluno no âmbito da sala de aula tradicional.

O oposto disso foi possível perceber nas aulas plugadas, onde essa assimetria fica minimizada e os alunos tem o professor (pelo menos para o formato conduzido nessa oficina) como uma pessoa em seu nível, que se mostra como um aprendiz junto ao aluno, por exemplo quando, juntos, professor e alunos tentaram concluir o nível 10 do labirinto do jogo Blockly Games, ou quando professor discute e faz sugestão aos alunos de "quantos passos aumentar" na flechinha do Scratch para que ela pudesse alcançar o alvo desejado.

Dito isso, reforçamos que esse formato híbrido da oficina nos permitiu observar que ele pode ser um formato promissor de trabalho, visto que as atividades, plugadas e desplugadas, operaram nesse contexto de maneira complementar e sinergética. Na medida em que o nível de complexidade das atividades foi avançando, os alunos foram explorando os diversos conceitos computacionais e potencializando o PC. Um trabalho exclusivamente desplugado, ou exclusivamente plugado, não traria tanta riqueza conforme observamos. Além disso, essa alternância de abordagens pode favorecer o atendimento a diferentes públicos, conforme ficou corroborado pelo "caso TDAH".

É importante destacar algumas contribuições que esse trabalho trouxe. Para mim, enquanto professor/pesquisador, reforçou a importância de estarmos sempre atentos a questões particulares dos alunos, diversificando os modos de trabalho, visto que cada um possui suas particularidades e que um formato de abordagem pode ser muito rico para um aluno enquanto para outro pode ser sem sentido; permitiu ainda observar a riqueza que há no trabalho com o PC e nas possibilidades de trabalhos interdisciplinares que podem ser desenvolvidos aliados ao PC, além, é claro, do amadurecimento enquanto pesquisador no contexto educacional.

Para os alunos, permitiu que observassem as tarefas de seus cotidianos como problemas que podem ser estruturados e resolvidos passo a passo; permitiu que tivessem uma noção da programação de computadores, conhecendo diversos conceitos computacionais e ainda os mostrou que são capazes de assumir um novo papel frente às TDIC, não mais de apenas consumidores, mas também construtores; permitiu que aperfeiçoassem e aprofundassem o pensamento estruturado do início da oficina, abrindo oportunidade para conhecerem diferentes

representações, seja no papel, seja no computador, enfim, permitiu em alguma medida a potencialização do PC desses alunos.

Em um contexto mais amplo, observamos nossa tese trazendo uma importante contribuição para o ensino/inserção do PC no contexto da Educação Básica. O formato híbrido e incremental proposto pela nossa oficina, bem como nosso olhar para o PC, pode servir de amparo para o desenvolvimento de um framework de ensino do PC na escola básica. Assim, vemos nosso trabalho trazendo mais uma perspectiva para o PC no contexto da Educação Básica.

Temos consciência de que nosso trabalho não "formou" programadores profissionais, e nem construímos grandes pensadores computacionais, mas certamente plantamos uma semente ao potencializarmos nesses alunos um estilo de pensamento utilizado por cientistas da computação, abrindo mais uma janela de oportunidade de estudos para eles. Levando em consideração a idade dos envolvidos, bem como o curto prazo da oficina aplicada, consideramos os resultados alcançados pelos alunos satisfatórios.

Estudos mais prolongados podem contribuir com a discussão trazida por essa tese, além de mudanças no formato da oficina. Aqui iniciamos trabalhando atividades desplugadas, e se tivéssemos iniciado com as atividades plugadas? E se ao invés de utilizarmos o *software* Scratch utilizássemos outro *software* de programação? E se não tivéssemos levado um problema pronto para ser resolvido pelo Scratch, mas deixássemos os alunos construírem um programa "livre", os mesmos conceitos computacionais e pilares seriam explorados? Como os pilares do PC poderiam ser explorados a partir da construção de artefatos físicos, utilizando, por exemplo, kits de robótica?

Muitas questões se colocam ao chegarmos nesse ponto da pesquisa e nos instigam, no futuro, a seguirmos olhando para essa região de inquérito tão rica e nova que é o Pensamento Computacional e o trabalho com as tecnologias digitais.

REFERÊNCIAS

ABBAGNANO, Nicola. **Dicionário de Filosofia**. 3. ed. São Paulo: Martins Fontes, 1998.

ALVES-MAZZOTTI, Alda Judith. Usos e abusos dos estudos de caso. **Cadernos de Pesquisa**, São Paulo, v. 26, n. 129, p. 637-651, dez. 2006.

ANDRÉ, Marli. Pesquisa em Educação: buscando rigor e qualidade. **Cadernos de Pesquisa,** São Paulo, n. 113, p. 51-64, jul. 2001.

ANDRÉ, Marli. O que é um estudo de caso qualitativo em educação? **Revista da Faeeba**, Salvador, v. 22, n. 40, p. 95-103, dez. 2013.

ARAÚJO, A. L. S. O.; ANDRADE, W. L.; GUERRERO, D. D. S.. Um Mapeamento Sistemático sobre a Avaliação do Pensamento Computacional no Brasil. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 5., 2016, Uberlândia. **Anais dos Workshops do V Congresso Brasileiro de Informática na Educação (CBIE 2016).** Uberlândia: Sbc, 2016. p. 1147-1158.

AVILA, C.; CAVALHEIRO, S.. Robótica Educacional como Estratégia de Promoção do Pensamento Computacional - Uma Proposta de Metodologia Baseada em Taxonomias de Aprendizagem. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 6., 2017, Recife. **Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação** (WCBIE 2017). Recife: Sbc, 2017. p. 1192-1201.

BARCELOS, T.; BORTOLETTO, R.; ANDRIOLI, M. G.. Formação online para o desenvolvimento do Pensamento Computacional em professores de Matemática. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 5., 2016, Uberlândia. **Anais dos Workshops do V Congresso Brasileiro de Informática na Educação (CBIE 2016).** Uberlândia: Sbc, 2016. p. 1228-1237.

BARCELOS, T. *et al*. Mensurando o desenvolvimento do Pensamento Computacional por meio de Mapas Auto-Organizáveis: um estudo preliminar em uma Oficina de Jogos Digitais. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 6., 2017, Recife. **Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação (WCBIE 2017).** Recife: Sbc, 2017. p. 932-941.

BARR, Valerie; STEPHENSON, Chris. Bringing Computational Thinking to K-12: what is involved and what is the role of the computer science education community?. **Acm Inroads**, [*s. l*], v. 2, n. 1, p. 48-54, mar. 2011.

BBC Bitesize. **What is computational thinking?** Disponível em <https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1>. Acesso em <02 fev. 2023>.

BERTAZINI, Enzo. **Pensamento computacional em livros didáticos do ensino médio**: sobre atividades e possibilidades. 2022. 225 f. Tese (Doutorado) - Curso de Educação Matemática, Universidade Anhanguera de São Paulo, São Paulo, 2022.

BERTO, Letícia Mara; ZAINA, Luciana Aparecida Martinez; SAKATA, Tiemi Christine. Metodologia Para Ensino do Pensamento Computacional para Crianças Baseada na Alternância

de Atividades Plugadas e Desplugadas. **Revista Brasileira de Informática na Educação – Rbie**, Porto Alegre, v. 27, n. 2, p. 1-22, maio 2019.

BICUDO, Maria Aparecida Viggiani. Pesquisa qualitativa e pesquisa qualitativa segundo a abordagem fenomenológica. In: BORBA, Marcelo de Carvalho; ARAÚJO, Jussara de Loiola (Org.). **Pesquisa Qualitativa em Educação Matemática.** 5. ed. Belo Horizonte: Autêntica, 2013. Cap. 4. p. 111-124.

BOGDAN, R.C.; BIKLEN, S.K. **Investigação qualitativa em Educação:** uma introdução à teoria e aos métodos. Porto: Porto Editora, 1999. 336p.

BRACKMANN, C. P.. **Desenvolvimento do Pensamento Computacional através de atividades desplugadas na Educação Básica**. 2017. 226 f. Tese (Doutorado) - Curso de Informática na Educação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular** – BNCC. Brasília, DF, 2018.

BRENNAN, Karen; RESNICK, Mitchel. New frameworks for studying and assessing the development of computational thinking. In: **Proceedings of the 2012 annual meeting of the American educational research association,** Vancouver, Canada. 2012. p. 25.

BUNDY, A. Computational Thinking is Pervasive. **Journal of Scientific and Practical Computing**, v. 1, p. 67–69, 2007.

CAMPANA, Samanta Bueno de Camargo. **O Pensamento Computacional na resolução de problemas na Matemática**: uma proposta de aplicação metodológica integrada na educação básica. 2022. 139 f. Tese (Doutorado) - Curso de Mídia e Tecnologia, Universidade Estadual Paulista Júlio de Mesquita Filho, Bauru, 2022.

CARVALHO, Felipe José Rezende de. Reflexões sobre o desenvolvimento de jogos com Scratch no ensino da Matemática. In: ENCONTRO NACIONAL DE EDUCAÇÃO MATEMÁTICA, 12., 2016, São Paulo. **Anais do 12º Encontro Nacional de Educação Matemática Universidade Cruzeiro do Sul**. São Paulo: Programa de Pós-graduação em Ensino de Ciências e Matemática da Universidade Cruzeiro do Sul, 2016. p. 1 – 12.

CARVALHO, Felipe José Rezende de. **Introdução à programação de computadores por meio de uma tarefa de Modelagem Matemática na Educação Matemática.** 2018. 133 f. Dissertação (Mestrado) - Curso de Ensino, Universidade Estadual do Oeste do Paraná, Foz do Iguaçu, 2018.

CARVALHO, F.; BRAGA, M. Pensamento Computacional na Educação Brasileira: um olhar segundo artigos do Congresso Brasileiro de Informática na Educação. **Revista Brasileira de Informática na Educação**, *[S. l.]*, v. 30, p. 237–261, 2022. DOI: 10.5753/rbie.2022.2649.

CASTELLS, Manuel. A Sociedade em Rede: do conhecimento à política. In: CASTELLS, Manuel; CARDOSO, Gustavo (org.). **A Sociedade em Rede Do Conhecimento à Acção Política**. Portugal: Imprensa Nacional, 2005. p. 17-30.

CHARÃO, A. *et al*. Computação e Psicologia: Construindo uma Parceria Interdisciplinar para o Ensino de Computação em Escolas. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 7., 2018, Fortaleza. **Anais dos Workshops do VII Congresso Brasileiro de Informática na Educação (WCBIE 2018).** Fortaleza: Sbc, 2018. p. 579-588.

FERREIRA, P. N. et al. O Uso da Robótica como Apoio à Alfabetização e à Introdução do Pensamento Computacional para Crianças. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 8., 2019, Brasília. **Anais dos Workshops do VIII Congresso Brasileiro de Informática na Educação** (WCBIE 2019). Brasília: Sbc, 2019. p. 238-247.

FERRO, Pedro. **Falta de profissionais de TI só tende a aumentar, dizem especialistas**. 2021. Disponível em: https://jornal.usp.br/atualidades/falta-de-profissionais-de-ti-so-tende-a-aumentar-dizem-especialistas/. Acesso em: 26 abr. 2022.

FINO, C. N. Matética e inovação pedagógica: o centro e a periferia. In: GOUVEIRA, F,; PEREIRA, M. G. (Org). **Didática e Matética**. Funchal: Universidade da Madeira, 2016. p.253-259

GARNICA, Antonio Vicente Marafioti. Algumas notas sobre Pesquisa Qualitativa e Fenomenologia. **Interface - Comunicação, Saúde e Educação,** Botucatu, v. 1, n. 1, p. 109-122, ago. 1997.

GEPETEC UNIRIO. **Roda de Conversa: Tecnologia Social GEPETEC Convida 2023**. YouTube, 3 de janeiro de 2023. Disponível em < https://www.youtube.com/watch?v=r-Ecp3MFFzM>. Acesso em <29 de mar. de 2023>.

GIL, Antônio Carlos. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2008.

GUARDA, Graziela Ferreira; PINTO, Sérgio Crespo Coelho da Silva. Dimensões do Pensamento Computacional: conceitos, práticas e novas perspectivas. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 9., 2020, Natal. **Anais do XXXI Simpósio Brasileiro de Informática na Educação.** Natal: Sbc, 2020. p. 1463-1472.

GUTIERREZ, Katerine Edith Tobio. **Um estudo sobre os estilos de aprendizagem, inteligência lógica matemática e Pensamento Computacional de alunos do Ensino Fundamental e médio de uma escola colombiana**. 2021. 165 f. Tese (Doutorado) - Curso de Educação Para A Ciência, Universidade Estadual Paulista Júlio de Mesquita Filho, Bauru, 2021.

HEMMENDINGER, David. A plea for modesty. **Acm Inroads**, [*s. l*], v. 2, n. 1, p. 4-7, jun. 2010.

HU, Chenglie. Computational thinking: what it might mean and what we might do about it. **Acm**, New York, p. 223-227, jan. 2011.

ISTE; CSTA. **Operational Definition of Computational Thinking for K–12 Education**. 2011. Disponível em: https://cdn.iste.org/www-root/Computational_Thinking_Operational_Definition_ISTE.pdf. Acesso em: 5 maio 2022.

KALELİOğLU, Filiz; GÜLBAHAR, Yasemin; KUKUL, Volkan. A Framework for Computational Thinking Based on a Systematic Research Review. **Baltic J. Modern Computing**, [*s. l*], v. 4, n. 3, p. 583-596, maio 2016.

LEÃO, Denise Maria Maciel. Paradigmas contemporâneas de educação: escola tradicional e escola construtivista. **Cadernos de Pesquisa**, São Paulo, n. 107, p. 187-206, jul. 1999.

LEÓN, Lucas Pordeus. **Brasil tem 152 milhões de pessoas com acesso à internet**. 2021. Disponível em: https://agenciabrasil.ebc.com.br/geral/noticia/2021-08/brasil-tem-152-milhoes-de-pessoas-com-acesso-internet. Acesso em: 16 mar. 2022.

LIUKAS, L. **Hello Ruby: adventures in coding**. Feiwel & Friends, 2015.

LUDKE, Menga; ANDRÉ, M. E. D. **Pesquisa em educação:** abordagens qualitativas. 2. ed. Rio de Janeiro: Anthares, 2018. 99p.

MALTEMPI, Marcus Vinicius. Construcionismo: pano de fundo para pesquisas em informática aplicada à Educação Matemática. In: BICUDO, Maria Aparedida Viggiani; BORBA, Marcelo de Carvalho (Org.). **Educação Matemática**: pesquisa em movimento. 4. ed. São Paulo: Cortez, 2012. Cap. 14. p. 287-307.

MARISTALAB. **Programação desplugada, o que é isso?** 2021. Disponível em: https://maristalab.com.br/praticas-educativas/programacao-desplugada-o-que-e-isso/. Acesso em: 23 mar. 2022.

MARQUIORI, Vânia Silvares. **Pensamento Computacional na compreensão de problemas do cotidiano feminino para o letramento em programação**. 2021. 104 f. Dissertação (Mestrado) - Curso de Educação Profissional e Tecnológica, Instituto Federal do Espírito Santo, Vitória, 2021.

MENEZES, Janaína. **Nos rastros de algoritmos pela cidade**: cartografia do desenvolvimento do pensamento computacional na perspectiva da educação onlife. 2022. 316 f. Tese (Doutorado) - Curso de Educação, Universidade do Vale dos Sinos, São Leopoldo, 2022.

MORENO-LEÓN, J.; ROBLES, G. Dr. Scratch: a web tool to automatically evaluate Scratch projects. 2015, London. **Proceedings of the 10th Workshop in Primary and Secondary Computing Education**. London: ACM, 2015. p. 132–133.

NAVARRO, Eloisa Rosotti. **O desenvolvimento do conceito de Pensamento Computacional na Educação Matemática segundo contribuições da teoria Histórico-Cultural**. 2021. 178 f. Tese (Doutorado) - Curso de Educação, Universidade Federal de São Carlos, São Carlos, 2021.

NUNES, Daltro José. **Ciência da Computação na Educação Básica**. 2011. Disponível em: http://gestaouniversitaria.com.br/artigos/ciencia-da-computacao-na-educacao-basica--3. Acesso em: 27 abr. 2022.

ORTIZ, Júlia Bathke; RAABE, André. Pensamento Computacional na Educação de Jovens e Adultos: lições aprendidas. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA

EDUCAÇÃO, 5., 2016, Uberlândia. **Anais dos Workshops do V Congresso Brasileiro de Informática na Educação.** Uberlândia: Sbc, 2016. p. 1087-1096.

PANCIERI, J. P.; SIQUEIRA, F. V.; OLIVEIRA, M. G. O Pensamento Computacional e a Robótica em Ações de Ressocialização de Jovens em Conflitos com a Lei. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 8., 2019, Brasília. **Anais dos Workshops do VIII Congresso Brasileiro de Informática na Educação** (WCBIE 2019). Brasília: Sbc, 2019. p. 268-277.

PAPERT, Seymour. **Logo:** Computadores e educação. São Paulo: Brasiliense, 1985.

PAPERT, Seymour. **A máquina das crianças:** repensando a escola na era da informática. Porto Alegre: Artmed, 1994. 210 p.

PONTE, João Pedro M. da. Estudos de Caso em Educação Matemática. **Bolema**, Rio Claro, v. 19, n. 25, p. 105-132, out. 2006.

RAABE, A. L. A. *et al*.. Um Instrumento para Diagnóstico do Pensamento Computacional. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 6., 2017, Recife. **Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação (WCBIE 2017).** Recife: Sbc, 2017. p. 1172-1181.

RESNICK, M. Point of View - Reviving Papert's Dream. **Educational Technology**, v.52, n.4, p.42-46, 2012.

SCHERER, Suely; FERNANDES, Frederico Fonseca. "Estar junto virtual ampliado" e o uso de tecnologias digitais em cursos de formação inicial de professores de matemática na modalidade de educação a distância. **Educação e Linguagens**, Campo Mourão, v. 3, n. 5, p.145-164, dez. 2014.

SELBY, Cynthia C.; WOOLLARD, John. Computational Thinking: the developing definition. In: ITICSE, 18., 2013, Canterbury. **Proceedings of the 18th ACM conference on Innovation and technology in computer science education.** England: University Of Kent, 2013. p. 1-6.

SCHORR, Maria Claudete. **PCOMP-MODEL**: desenvolvendo o pensamento computacional na educação básica para auxiliar na aprendizagem de algoritmos e programação do ensino superior. 2020. 190 f. Tese (Doutorado) - Curso de Informática na Educação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2020.

SILVA, Leonardo José da. **Educação Matemática e ambientes virtuais**: uma análise das interações ocorridas em um chat educacional. 2013. 300 f. Tese (Doutorado) - Curso de Educação Para A Ciência, Universidade Estadual Paulista, Bauru, 2013.

SILVA, Leonardo José da; CARVALHO, Felipe José Rezende de. Pensando a robótica na Educação Básica. **Ridema**, Juiz de Fora, v. 2, n. 1, p.137-159, jun. 2018.

SILVA, D. P. et al.. Aplicação de Robótica na Educação de Forma Gradual para o Estímulo do Pensamento Computacional. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 5., 2016, Uberlândia. **Anais dos Workshops do V Congresso Brasileiro de Informática na Educação** (CBIE 2016). Uberlândia: Sbc, 2016. p. 1188-1197.

SOUZA, Joelson Carvalho; HICKMANN, Adolfo Antonio; ASINELLI-LUZ, Araci; HICKMANN, Girlane Moura. A influência das emoções no aprendizado de escolares. **Revista Brasileira de Estudos Pedagógicos**, [S.L.], v. 101, n. 258, p. 382-403, ago. 2020. Instituto Nacional de Estudos e Pesquisas Educacionais Anisio Teixeira. http://dx.doi.org/10.24109/2176-6681.rbep.101i258.4279.

UNIVESP. **Pensamento Computacional**: Introdução. YouTube, 8 de agosto de 2022. Disponível em: <https://www.youtube.com/watch?v=s9Q7Ik_sle4>. Acesso em: 31 de agosto de 2022.

VALENTE, José Armando. **A Espiral da Espiral de Aprendizagem:** o processo de compreensão do papel das tecnologias de informação e comunicação na educação. 2005. 232 f. Tese (Doutorado) - Curso de Professor Livre Docente, Instituto de Artes, Universidade Estadual de Campinas, Campinas, 2005a.

VALENTE, José Armando. **Informática na Educação**: instrucionismo x construcionismo. instrucionismo X construcionismo. 2005b. Disponível em: https://educacaopublica.cecierj.edu.br/artigos/2/1/informaacutetica-na-educaccedilatildeo-instrucionismo-x-construcionismo. Acesso em: 22 mar. 2023.

VICARI, Rosa Maria; MOREIRA, Álvaro; MENEZES, Paulo Blauth. **Pensamento Computacional:** revisão bibliográfica. Porto Alegre: Ufrgs, 2018.

WING, J. M.. Computational Thinking. **Communications Of The Acm**, New York, v. 49, n. 3, p. 33-35, mar. 2006.

WING, J. M.. **Computational Thinking Benefits Society**. 2014. Disponível em: http://socialissues.cs.toronto.edu/index.html%3Fp=279.html. Acesso em: 28 abr. 2022.

WING, J. M.. Computational thinking's influence on research and education for all. **Italian Journal Of Educational Technology**, [*s. l*], v. 25, n. 2, p. 7-14, jan. 2017.

ZANETTI, H.; BORGES, M.; RICARTE, I. Pensamento Computacional no Ensino de Programação: Uma Revisão Sistemática da Literatura Brasileira. In: Simpósio Brasileiro De Informática Na Educação (SBIE). **Anais** [...], 2016, p. 21-30.

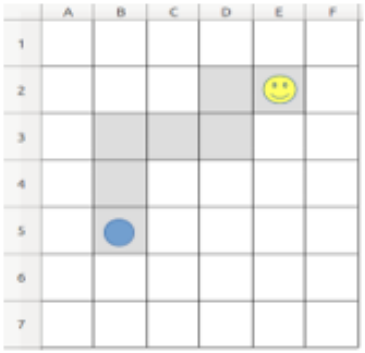
ZANETTI, H. U. P.; OLIVEIRA, C. L. V.. Prática de ensino de Programação de Computadores com Robótica Pedagógica e aplicação de Pensamento Computacional. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 4., 2015, Maceió. **Anais dos Workshops do IV Congresso Brasileiro de Informática na Educação** (CBIE 2015). Maceió: Sbc, 2015. p. 1236-1245.

APÊNDICE A – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS - AULA 2

PROGRAMADOR

M1

Olhe para esse desenho e **diga** ao computador onde o círculo azul está e depois quais os passos (pra frente, pra baixo, pra cima, pra direita) o círculo deve dar até o rostinho feliz.



COMPUTADOR

M1

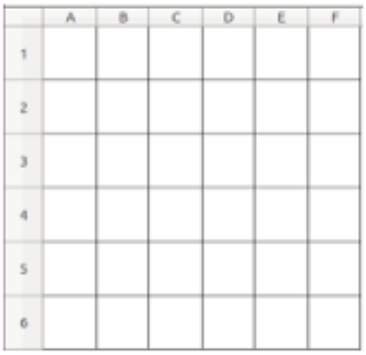
Anote primeiro a posição do círculo azul e depois os passos que devem ser dados (utilize setas para registrar os passos ← ↑ → ↓)

_____

_____

_____

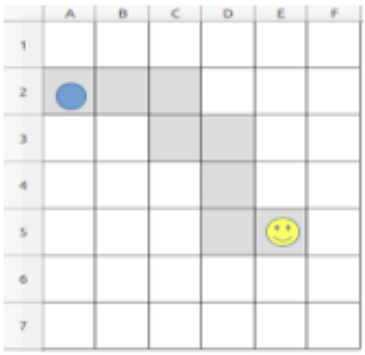
Agora desenhe o círculo, o rosto feliz e pinte o caminho que os conecta.



PROGRAMADOR

M2

Olhe para esse desenho e **diga** ao computador onde o círculo azul está e depois quais os passos (pra frente, pra baixo, pra cima, pra direita) o círculo deve dar até o rostinho feliz.



COMPUTADOR

M2

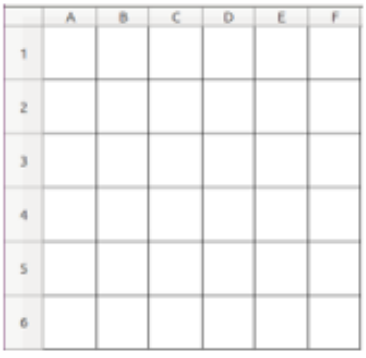
Anote primeiro a posição do círculo azul e depois os passos que devem ser dados (utilize setas para registrar os passos ← ↑ → ↓)

_____

_____

_____

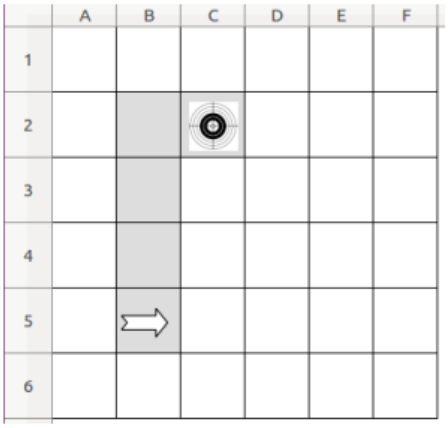
Agora desenhe o círculo, o rosto feliz e pinte o caminho que os conecta.



PROGRAMADOR

M3 1

Olhe para esse desenho e **diga** ao computador onde a flecha está e depois quais os comandos para ela chegar ao alvo. ATENÇÃO: a flecha só "anda" na direção de sua ponta, então ela precisará girar - para direita ou esquerda - antes de "andar".



COMPUTADOR

M3 1

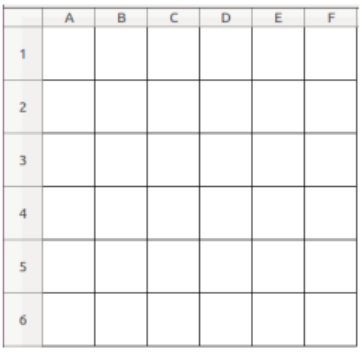
Anote primeiro a posição da flecha e depois seus movimentos. Utilize setas para registrar os movimentos. Detalhe: utilize ↻ para girar à direita e ↺ para girar à esquerda.

_____

_____

_____

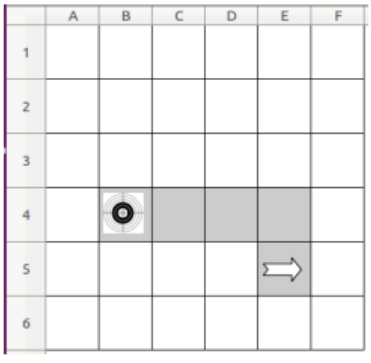
Agora desenhe a flecha, o alvo e pinte o caminho que os conecta.



PROGRAMADOR

M3 2

Olhe para esse desenho e **diga** ao computador onde a flecha está e depois quais os comandos para ela chegar ao alvo. ATENÇÃO: a flecha só "anda" na direção de sua ponta, então ela precisará girar - para direita ou esquerda - antes de "andar".



COMPUTADOR

M3 2

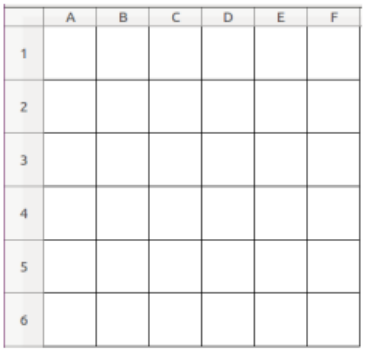
Anote primeiro a posição da flecha e depois seus movimentos. Utilize setas para registrar os movimentos. Detalhe: utilize ↻ para girar à direita e ↺ para girar à esquerda.

_____

_____

_____

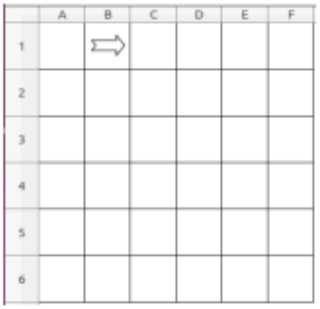
Agora desenhe a flecha, o alvo e pinte o caminho que os conecta.



APÊNDICE B – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS – AULA 4 PARTE INICIAL

Onde devemos colocar o alvo para que a flecha o acerte? Pinte o caminho formado pelo algoritmo e desenhe o alvo.



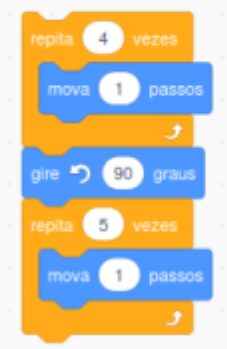


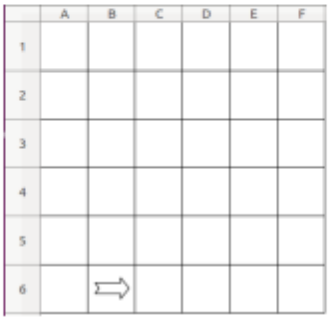
Onde devemos colocar o alvo para que a flecha o acerte? Pinte o caminho formado pelo algoritmo e desenhe o alvo.



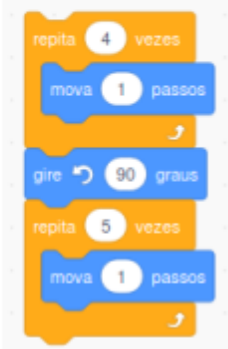


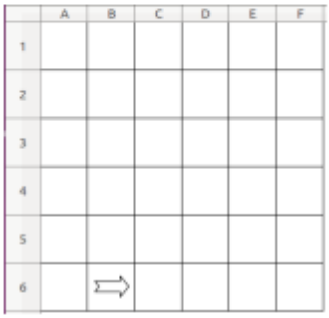
Onde devemos colocar o alvo para que a flecha o acerte? Pinte o caminho formado pelo algoritmo e desenhe o alvo.





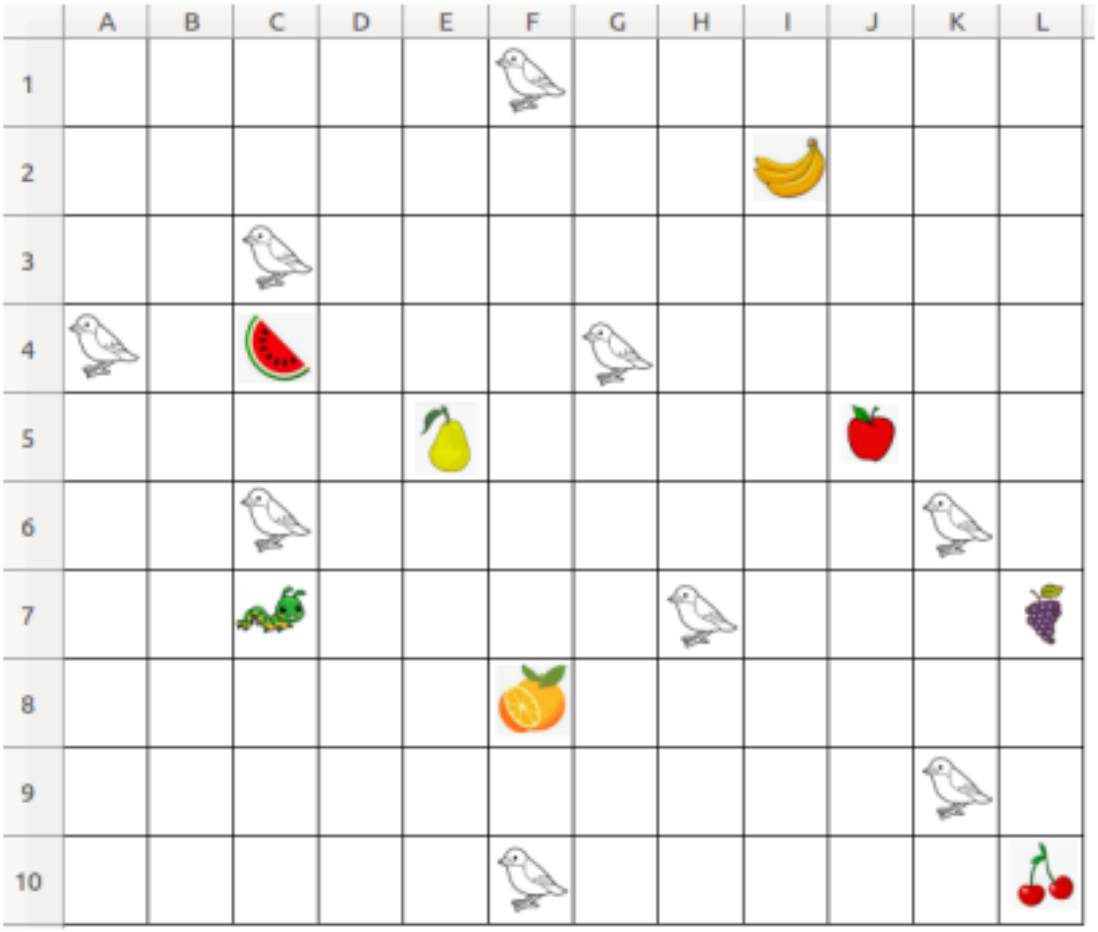
Onde devemos colocar o alvo para que a flecha o acerte? Pinte o caminho formado pelo algoritmo e desenhe o alvo.





APÊNDICE C – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS – AULA 4 PARTE FINAL - 1ª APLICAÇÃO

Vamos ajudar a lagarta comilona a se alimentar das frutas, mas cuidado com o pássaro predador. Lembre-se que a lagarta não anda na diagonal.



1) Escreva uma sequência de passos utilizando setas para ajudar a lagarta a:

a) comer a maçã

b) comer a melancia e em seguida a pera

c) comer a banana e em seguida a uva

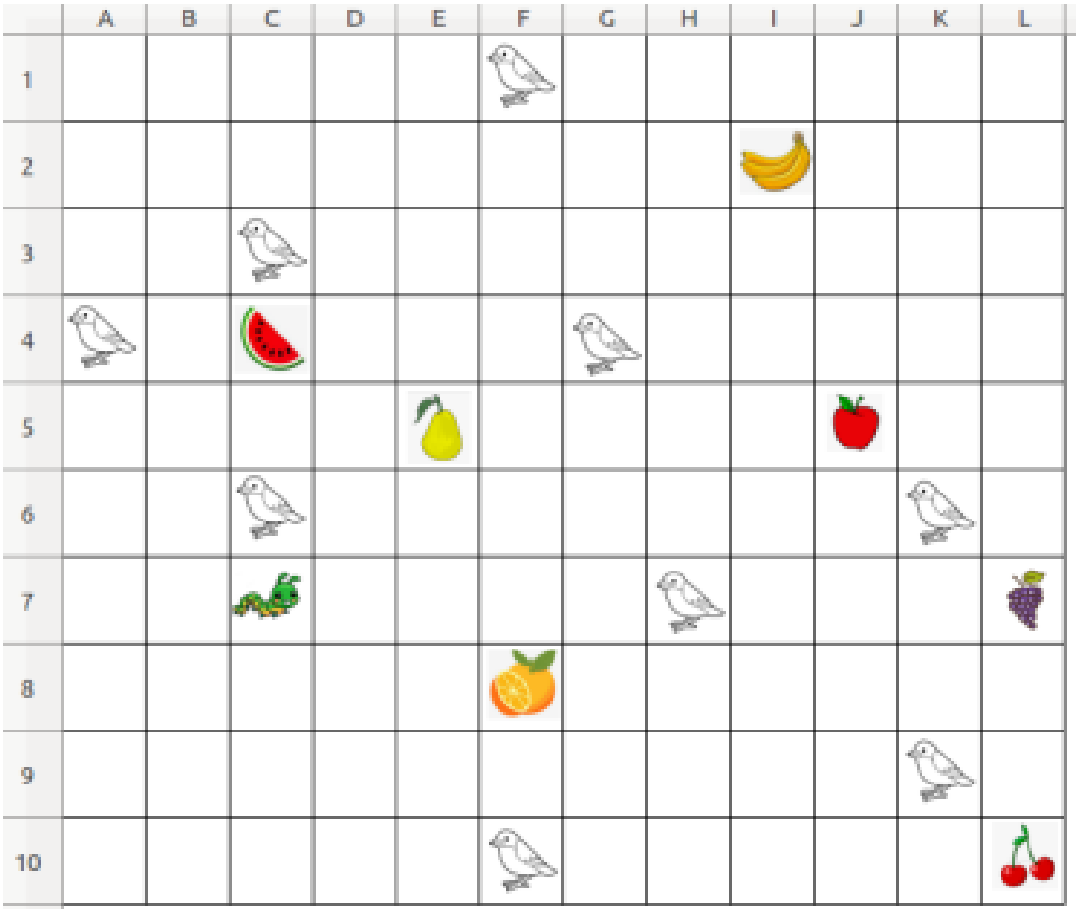
d) comer as cerejas e em seguida a laranja

Suas soluções utilizaram o menor número de comandos possível? Reescreva suas soluções agrupando comandos iguais quando possível.

2) Um programador quer levar a lagarta para comer a pera, a maçã e a uva, nessa ordem. Para isso criou a seguinte sequência de passos: 2 x (→ ↑) 5 x (→) ↓ 2 x (→) ↓
Verifique a programação e, caso seja necessário, faça ajustes para alcançar seu objetivo.

APÊNDICE D – ENUNCIADOS DAS ATIVIDADES DESPLUGADAS – AULA 4 PARTE FINAL - 2ª APLICAÇÃO

Vamos ajudar a lagarta comilona a se alimentar das frutas, mas cuidado com o pássaro predador. Lembre-se que a lagarta não anda na diagonal.



Comandos a serem utilizados para guiar a lagarta:



1) Utilize quantos comandos precisar para ajudar a lagarta a:

a) comer a maçã

c) comer a banana e em seguida a uva

b) comer a melancia e em seguida a pera

d) comer as cerejas e em seguida a laranja

2) Um programador criou um algoritmo para levar a lagarta para comer a pera, a maçã e a uva, nessa ordem. Faça a correção do algoritmo apresentado.



APÊNDICE E – DISCUSSÃO INICIAL PARA OS PROJETOS SCRATCH

Introdução aos problemas das equipes

O diretor da Organização Mundial de Saúde (OMS), Tedros Adhanom, decretou pandemia do novo coronavirus em 11/03/2020. A partir dessa declaração os países precisaram tomar medidas duras para tentar conter a propagação do vírus.

A partir dessa data a população se torna mais alerta e tem seu estilo de vida totalmente modificado. Atividades simples de sair com os amigos para ir ao shopping ou ir até uma lanchonete já não seriam mais possíveis, visto que esses ambientes aglomeram pessoas e esse é o maior vilão para a propagação do vírus.

Além da mudança no estilo de vida das pessoas, a realidade dos hospitais também muda drasticamente. É necessário se prepararem para receber pacientes. É preciso mais verba para adquirir máscaras, álcool; é preciso aumentar leitos para que tenham capacidade de atender mais pessoas.

Cada uma das equipes irá ajudar na resolução dos problemas enfrentados pelo setor de saúde dentro desse contexto. As equipes deverão desenvolver programas utilizando o Scratch para auxiliar na solução dessas questões.

APÊNDICE F – PROBLEMAS COVID PARA RESOLVER NO SCRATCH

Vacina

Programa 1: Programa Scratch deverá apresentar por quanto tempo um estoque de vacina irá durar.
- O que o programa precisa "saber" para dar essa resposta?
a) quantidade de vacinas em estoque
b) quantidade de vacinas usadas por dia
- Qual a funcionalidade que o programa deverá apresentar?
a) ao ser executado deverá "perguntar" ao usuário qual é o estoque e quantas vacinas são usadas por dia
b) com as informações armazenadas o programa faz o cálculo e apresenta o resultado em dias

Um avanço nesse programa: inserir uma condicional.
Quando o estoque estiver baixo (sua duração for menor que 7 dias, por exemplo) deverá emitir um "alerta" dizendo que é necessário repor estoque.

Programa 2: Criar uma animação em que um objeto (vacina) se movimenta na tela a partir das setas do teclado até chegar a um paciente (outro objeto). Quando a vacina encostar no paciente emitir a mensagem de que o paciente foi vacinado. (aqui é interessante deixar fluir a criatividade dos alunos para que eles criem algo semelhante a um game, pontuação, obstáculos etc)

ANOTAÇÕES

O que será necessário para resolver esse problema? Como será o programa? Registre aqui esses requisitos.

_____

_____

_____

_____

_____

_____

Álcool

Programa 1: Programa Scratch deverá apresetar por quanto tempo um estoque de álcool de determinado hospital irá durar.
- O que o programa precisa "saber" para dar essa resposta?
a) quantidade de frascos (litros) de álcool em estoque
b) quantidade de álcool (litros) usados por dia
- Qual a funcionalidade que o programa deverá apresentar?
a) ao ser executado deverá "perguntar" ao usuário qual é o estoque de álcool e quantos litros são usados por dia
b) com as informações armazenadas o programa faz o cálculo e apresenta o resultado em dias

Um avanço nesse programa: inserir uma condicional.
Quando o estoque estiver baixo (sua duração for menor que 7 dias, por exemplo) deverá emitir um "alerta" dizendo que é necessário repor estoque.

Programa 2: Criar uma animação em que um objeto (álcool) se movimenta na tela a partir das setas do teclado até chegar a um hospital (outro objeto). Quando o álcool encostar no hospital emitir a mensagem de que o estoque foi abastecido. (aqui é interessante deixar fluir a criatividade dos alunos para que eles criem algo semelhante a um game, pontuação, obstáculos etc)

ANOTAÇÕES

O que será necessário para resolver esse problema? Como será o programa? Registre aqui esses requisitos.

_____

_____

_____

_____

_____

_____

_____

Máscaras

Programa 1: Programa Scratch deverá apresentar o valor gasto por um hospital com máscaras

- O que o programa precisa "saber" para dar essa resposta?
a) quantidade de máscaras usadas por dia
b) valor de cada máscara
- Qual a funcionalidade que o programa deverá apresentar?
a) ao ser executado deverá "perguntar" ao usuário qual é o quantidade de máscaras usadas ao dia e qual o valor de cada máscara
b) com as informações armazenadas o programa faz o cálculo e apresenta o valor

Um avanço nesse programa: inserir uma condicional.
Quando o valor estiver muito alto (superar R$500,00, por exemplo) deverá emitir um "alerta" dizendo que há um consumo excessivo do material.

Programa 2: Criar uma animação em que um objeto (máscara) se movimenta na tela a partir das setas do teclado até chegar a um médico (outro objeto). Quando a máscara encostar no médico emitir a mensagem de que ele está protegido. (aqui é interessante deixar fluir a criatividade dos alunos para que eles criem algo semelhante a um game, pontuação, obstáculos etc)

ANOTAÇÕES

O que será necessário para resolver esse problema? Como será o programa? Registre aqui esses requisitos.

_____

_____

_____

_____

_____

_____

_____

Leitos de hospital

Programa 1: Programa Scratch deverá apresentar o tempo, em dias, em que o hospital ficará cheio

- O que o programa precisa "saber" para dar essa resposta?
a) quantidade de leitos disponíveis
b) quantidade de pacientes internados por dia (como simplificação, considerar que os pacientes não terão alta no período analisado)
- Qual a funcionalidade que o programa deverá apresentar?
a) ao ser executado deverá "perguntar" ao usuário qual é o quantidade de leitos do hospital e quantos pacientes são internados por dia
b) com as informações armazenadas o programa faz o cálculo e apresenta o valor

Um avanço nesse programa: inserir uma condicional.
Quando o valor estiver muito baixo (inferior a 3 dias, por exemplo) deverá emitir um "alerta" dizendo que o hospital estará lotado em breve.

Programa 2: Criar uma animação em que um objeto (paciente) se movimenta na tela a partir das setas do teclado até chegar a um hospital (outro objeto). Quando o paciente encostar no hospital emitir a mensagem de que ele foi internado. (aqui é interessante deixar fluir a criatividade dos alunos para que eles criem algo semelhante a um game, pontuação, obstáculos etc)

ANOTAÇÕES

O que será necessário para resolver esse problema? Como será o programa? Registre aqui esses requisitos.

_____

_____

_____

_____

_____

_____

_____

Quantidade de pessoas infectadas

Programa 1: Programa Scratch deverá apresentar o tempo, em dias, em que o pico da pandemia irá ocorrer

- O que o programa precisa "saber" para dar essa resposta?
a) número máximo de infectados (pico)
b) quantidade de pessoas que estão se infectando por dia
- Qual a funcionalidade que o programa deverá apresentar?
a) ao ser executado deverá "perguntar" ao usuário qual é o número de pessoas infectadas que representa o pico da pandemia e o número de pessoas que se infectam por dia
b) com as informações armazenadas o programa faz o cálculo e apresenta o valor

Um avanço nesse programa: inserir uma condicional.
Quando o valor estiver abaixo de 3 dias, por exemplo, o programa deverá emitir um "alerta" dizendo que o pico da pandemia está próximo

Programa 2: Criar uma animação em que um objeto (vírus) se movimenta na tela a partir das setas do teclado até chegar a uma pessoa (outro objeto). Quando o vírus encostar na pessoa emitir a mensagem de que ele foi infectado. (aqui é interessante deixar fluir a criatividade dos alunos para que eles criem algo semelhante a um game, pontuação, obstáculos etc)

ANOTAÇÕES

O que será necessário para resolver esse problema? Como será o programa? Registre aqui esses requisitos.

_____

_____

_____

_____

_____

_____

_____

APÊNDICE G – ROTEIRO DA ENTREVISTA COLETIVA

Entrevista coletiva
Fechamento da oficina

- Discutir novamente sobre o PC. O que é o PC?

- Na nossa segunda aula fizemos aquela brincadeira de um aluno programar o outro, dizer o caminho a ser percorrido no quadriculado. O que foi mais difícil pra vocês nessa atividade?

- Vocês conseguem enxergar alguma relação da aula em que um programou o outro com a que a gente jogou o blockly games (labirinto)?

- Os jogos blockly games ajudaram a entender a questão de virar pra direita/esquerda?

- Resolver o labirinto da lagarta comilona (no papel) foi mais fácil ou mais difícil do que resolver os labirintos do blockly games? (estimular os alunos a argumentarem sobre suas respostas)

- Vocês conseguem relacionar alguma coisa das aulas anteriores ao projeto Scratch sobre a COVID? (ir fomentando a conversa falando sobre alguns códigos utilizados nas aulas desplugadas/plugadas etc)

- Alguém teve curiosidade de estudar um pouco mais sobre o Scratch depois que começamos nosso projeto?

- O que vocês acharam de trabalhar em grupo? Vocês pediram ajuda para os colegas?

- Qual sugestão vocês dariam para melhorar a oficina?

APÊNDICE H – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

CEFET/RJ

TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO AOS PAIS E RESPONSÁVEIS PELOS ALUNOS MENORES DE IDADE

Convidamos seu filho(a) (ou o menor sob sua responsabilidade) para participar de um estudo sobre redes de conhecimento e aquisição de competências através de projetos colaborativos, que faz parte do projeto "CONSTRUINDO REDES DE CONHECIMENTO ATRAVES DE PROJETOS COLABORATIVOS", do Programa de Pós-graduação em Ciência, Tecnologia e Educação do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ. Para este estudo serão convidados todos os alunos que participarem da oficina "Desenvolvendo o Pensamento Computacional por meio de uma abordagem hìbrida". O objetivo da pesquisa é verificar o desenvolvimento do Pensamento Computacional por meio da resolução de problemas.

O estudo ocorrerá durante a participação dos estudantes nesta oficina que acontecerá no Cap João XXIII em algumas aulas de matemática. A presença na oficina é voluntária, e caso seu(ua) filho(a) não deseje participar não sofrerá nenhuma punição. A participação na pesquisa também é voluntária, ou seja, seu(ua) filho(a) poderá participar da oficina e não participar da pesquisa, que também não sofrerá qualquer tipo de punição.

Caso seu(ua) filho(a) concorde em participar da pesquisa, terá suas atividades observadas pelo pesquisador. Seu(ua) filho(a) não precisa saber responder as questões, não havendo compromisso de acertos, por isso não se identificará nas atividades.

A oficina será filmada e algumas conversas dos alunos serão gravadas para efetuarmos análise da interação dos alunos durante as atividades. Os alunos poderão se sentir acanhados/constrangidos devido às gravações, porém, reiteramos que essas gravações serão utilizadas exclusivamente para estudo científico das atividades, não sendo vinculada a qualquer meio de comunicação. Publicações científicas ocorridas com esses dados serão anônimas, ou seja, o nome/imagem do seu(ua) filho (a) não serão divulgados. Os resultados estarão à sua disposição quando finalizada a pesquisa. Os dados e instrumentos da pesquisa ficarão arquivados por um período de 5 (cinco) anos e após este tempo serão destruídos.

Para participar não há nenhum custo e também nenhuma remuneração, entretanto, a participação do seu(ua) filho(a) (ou menor sob sua responsabilidade) será muito importante, pois esperamos que futuramente os resultados deste estudo sejam usados em benefício de outras pessoas e/ou na produção de conhecimento científico sobre o tema, como para que os pesquisadores possam avaliar como as redes de conhecimento se formam e a validade de projetos colaborativos para a aquisição de competências específicas.

Esta atividade não será utilizada como teste ou prova. Caso o aluno(a) não consiga responder as questões, isso não acarretará prejuízos, nem constrangimentos, até porque tudo ocorrerá de forma anônima.

Então, se está claro para o senhor(a), peço que assine e rubrique este documento. Se o senhor(a) não assinar, seu(ua) filho(a) ou menor sob sua responsabilidade não poderá participar da pesquisa, embora possa participar da oficina de trabalho, sem qualquer problema. Caso você tenha alguma dúvida, haverá possibilidade de conversas adicionais com a equipe, caso o menor ou o responsável solicite, podendo entrar em contato com o pesquisador Marco Braga, no endereço CENTRO FEDERAL DE EDUCAÇAO TECNOLOGICA CELSO SUCKOW DA FONSECA (CEFET/RJ), Avenida Maracanã 229, Maracanã, Bloco E, 5º andar, sala 503.12, pelo telefone (21) 2566-3179, e-mail: marcobraga@namalab.education, ou poderá entrar em contato com o Comitê de Etica em Pesquisa com Seres Humanos do Instituto Oswaldo Cruz. O Comitê de Etica é a instância que tem por objetivo defender os interesses dos participantes da pesquisa em sua integridade e dignidade e para contribuir no desenvolvimento da pesquisa dentro de padrões éticos. Dessa forma, o comitê tem o papel de avaliar e monitorar o andamento do projeto, de modo que a pesquisa respeite os princípios éticos de proteção aos direitos humanos, da dignidade, da autonomia, da não maleficência, da confidencialidade e da privacidade. Avenida Brasil, 4.036, sala: 705, Manguinhos, RJ-CEP 21.040-360, telefone (21) 3882-9011, e-mail: cepfiocruz@ioc.fiocruz.br.

O seu consentimento deve estar de acordo com o consentimento do seu(ua) filho(a) (ou o menor sob sua responsabilidade), isto é, se seu(ua) filho(a) (ou o menor sob sua responsabilidade) não concordar em participar, a opinião dele(a) será respeitada.

Este Termo está sendo apresentado em duas vias de igual teor, sendo que uma ficará com o pesquisador e a outra ficará com o sr(a).

Meus sinceros agradecimentos por sua colaboração.

CEFET/RJ

AUTORIZAÇÃO

Eu, _____, CONCORDO com a participação do(a) menor _____sob minha responsabilidade na pesquisa "CONSTRUINDO REDES DE CONHECIMENTO ATRAVÉS PROJETOS COLABORATIVOS".

Esclareço que fui informado(a) dos objetivos do presente estudo de maneira clara e detalhada e esclareci minhas dúvidas. Sei que a qualquer momento poderei solicitar novas informações e modificar a decisão do(a) menor sob minha responsabilidade de participar, se assim desejar. Recebi via de igual teor deste termo de consentimento livre e esclarecido e me foi dada a oportunidade de ler e esclarecer minhas dúvidas.

Em ____ de _____ de 2022.

_____
Assinatura do(a) responsável

_____
p/ Marco Braga
Coordenador da pesquisa

APÊNDICE I – TERMO DE ASSENTIMENTO LIVRE E ESCLARECIDO

CEFET/RJ

TERMO DE ASSENTIMENTO LIVRE E ESCLARECIDO AOS MENORES DE IDADE

Você está sendo convidado para participar de uma pesquisa que faz parte do projeto "CONSTRUINDO REDES DE CONHECIMENTO ATRAVÉS DE PROJETOS COLABORATIVOS", do Programa de Pós-graduação em Ciência, Tecnologia e Educação do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ. Faremos uma oficina chamada "Desenvolvendo o Pensamento Computacional por meio de uma abordagem híbrida" no âmbito das aulas de matemática, objetivando verificar como vocês (alunos) se agrupam e trocam informações para a resolução de problemas que estimulem o Pensamento Computacional. Com isto poderemos avaliar se este tipo oficina serve para ajudar os estudantes a aprenderem de maneira mais divertida, e possam ser utilizados como material de apoio.

Caso você não queira participar, não tem problema, você não será punido por isso. Caso deseje participar, você estará apto a nos ajudar em nossa pesquisa. Você pode participar da oficina e não participar da pesquisa, que também nada acontecerá com você.

Durante a oficina, você e seus colegas serão filmados e observados pelo pesquisador e terão algumas conversas gravadas e, por esse motivo, poderão se sentir acanhados, porém, reiteramos que essas imagens e áudios serão utilizados exclusivamente pelo pesquisador para verificar a interação entre vocês, não sendo vinculados em nenhuma forma de comunicação ao público.

Todo este exercício não será um teste ou prova, tanto que você não colocará seu nome nele. Se você não souber responder às perguntas não terá nenhum problema e nem isso será utilizado para avaliação da sua escola.

Para participar, você não precisa pagar nada e nem receberá nada, mas a sua ajuda será muito importante para nós. Esperamos que futuramente os resultados deste estudo sejam usados em benefício de outras pessoas e/ou na produção de conhecimento científico sobre o tema.

Então, se você entendeu tudo o que está escrito, peço que assine este documento. Se você não assinar, não poderá participar da pesquisa, mas poderá participar da. Caso você tenha alguma dúvida, pode conversar com a equipe, entrando em contato com o pesquisador Marco Braga no endereço CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA (CEFET/RJ), Avenida Maracanã 229, Maracanã, Bloco E, 5º andar, sala 503.12, pelo telefone (21) 2566-3179, e-mail: marcobraga@namelab.education, ou com o Comitê de Ética em Pesquisa com Seres Humanos do Instituto Oswaldo Cruz – CEP/IOC. O Comitê de Ética é a instância que tem por objetivo defender os interesses dos participantes da pesquisa em sua integridade e dignidade e para contribuir no desenvolvimento da pesquisa dentro de padrões éticos. Dessa forma, o comitê tem o papel de avaliar e monitorar o andamento do projeto, de modo que a pesquisa respeite os princípios éticos de proteção aos direitos humanos, da dignidade, da autonomia, da não maleficência, da confidencialidade e da privacidade e fica na Avenida Brasil, 4.036, sala: 705, Manguinhos, RJ-CEP 21.040-360, telefone (21) 3882-9011, e-mail: cepfiocruz@ioc.fiocruz.br.

O seu consentimento deve estar de acordo com o consentimento dos seus pais ou responsáveis, que se não concordarem com a sua participação, a opinião deles será respeitada e você não poderá tomar parte.

Desde já, agradecemos a sua colaboração e solicitamos a sua assinatura e rubrica de autorização neste termo, que será também assinado pelo pesquisador responsável em duas vias iguais, sendo que uma ficará com você e outra com o pesquisador.

Meus sinceros agradecimentos por sua colaboração.

CEFET/RJ

AUTORIZAÇÃO

Eu, _____, CONCORDO em participar da pesquisa "CONSTRUINDO REDES DE CONHECIMENTO ATRAVÉS PROJETOS COLABORATIVOS". Fui informado(a) dos objetivos do presente estudo de maneira clara e detalhada e esclareci minhas dúvidas. Sei que a qualquer momento poderei pedir novas informações e modificar minha decisão de participar, se assim desejar, sem nenhum problema. Recebi via de igual teor deste termo de assentimento livre e esclarecido e me foi dada a oportunidade de ler e esclarecer minhas dúvidas.

Em ____ de _____ de 2021.

_____

Assinatura do(a) participante da pesquisa

_____

p/ Marco Braga
Coordenador da pesquisa