

**UNIVERSIDADE ESTADUAL PAULISTA “JULIO DE MESQUITA FILHO”
FACULDADE DE ARQUITETURA, ARTES, COMUNICAÇÃO E DESIGN
PROGRAMA DE PÓS-GRADUAÇÃO EM MÍDIA E TECNOLOGIA
DOUTORADO ACADÊMICO**

SAMANTA BUENO DE CAMARGO CAMPANA

**O PENSAMENTO COMPUTACIONAL NA
RESOLUÇÃO DE PROBLEMAS NA MATEMÁTICA:
Uma proposta de aplicação metodológica integrada na Educação Básica**

Bauru, SP

2022

SAMANTA BUENO DE CAMARGO CAMPANA

**O PENSAMENTO COMPUTACIONAL NA
RESOLUÇÃO DE PROBLEMAS NA MATEMÁTICA:
Uma proposta de aplicação metodológica integrada na Educação Básica**

Tese apresentada à Faculdade de Arquitetura, Artes, Comunicação e Design da Universidade Estadual Paulista “Júlio de Mesquita Filho” para obtenção do título de Doutora em Mídia e Tecnologia pelo Programa de Pós-graduação em Mídia e Tecnologia – Doutorado Acadêmico, na área de Concentração de Ambientes Midiáticos e Tecnológicos, Linha de Pesquisa de Tecnologias Midiáticas, sob orientação do Prof. Assoc. Eduardo Martins Morgado e Coorientação do Prof. Dr. Wilson Massashiro Yonezawa.

Bauru, SP

2022

Campana, Samanta Bueno de Camargo.
O PENSAMENTO COMPUTACIONAL NA
RESOLUÇÃO DE PROBLEMAS NA MATEMÁTICA:
Uma proposta de aplicação metodológica integrada
na Educação Básica / Samanta Bueno de Camargo
Campana, 2022.
142 f. : il.
Orientador: Eduardo Martins Morgado.
Coorientador: Wilson Massashiro Yonezawa.
Tese (Doutorado)– Universidade Estadual
Paulista (Unesp). Faculdade de Arquitetura,
Artes e Comunicação, Bauru, 2022.

ATA DA DEFESA PÚBLICA DA TESE DE DOUTORADO DE SAMANTA BUENO DE CAMARGO CAMPANA, DISCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM MÍDIA E TECNOLOGIA, DA FACULDADE DE ARQUITETURA, ARTES, COMUNICAÇÃO E DESIGN - CÂMPUS DE BAURU.

Aos 30 dias do mês de setembro do ano de 2022, às 14:30 horas, por meio de Videoconferência, realizou-se a defesa de TESE DE DOUTORADO de SAMANTA BUENO DE CAMARGO CAMPANA, intitulada **O Pensamento Computacional na Resolução de Problemas na Matemática: uma proposta de aplicação metodológica integrada na Educação Básica**. A Comissão Examinadora foi constituída pelos seguintes membros: Professor Associado EDUARDO MARTINS MORGADO (Orientador(a) - Participação Virtual) do(a) Departamento de Computação da Faculdade de Ciências / Universidade Estadual Paulista , Professora Doutora KETILIN MAYRA PEDRO (Participação Virtual) do(a) Departamento de Psicologia Centro de Educação e Ciências Humanas / Universidade Federal de São Carlos - UFSCar, Professora Doutora MYRIAN LUCIA RUIZ CASTILHO (Participação Virtual) do(a) Educação / Universidade de Marília. Após a exposição pela doutoranda e arguição pelos membros da Comissão Examinadora que participaram do ato, de forma presencial e/ou virtual, a discente recebeu o conceito final: APROVADA. Nada mais havendo, foi lavrada a presente ata, que após lida e aprovada, foi assinada pelo(a) Presidente(a) da Comissão Examinadora.

A handwritten signature in black ink, appearing to read "Eduardo", is placed above the name of the signatory.

Professor Associado EDUARDO MARTINS MORGADO

Tese de Doutorado de Samanta Bueno de Camargo Campana, sob o título “**O Pensamento Computacional na Resolução de Problemas na Matemática: uma proposta de aplicação metodológica integrada na Educação Básica**”, apresentada à Faculdade de Arquitetura, Artes, Comunicação e Design da Universidade Estadual Paulista “Júlio de Mesquita Filho”, para obtenção do título de Doutora em Mídia e Tecnologia pelo Programa de Pós-graduação em Mídia e Tecnologia – Doutorado Acadêmico, na área de Concentração de Ambientes Midiáticos e Tecnológicos, Linha de Pesquisa de Tecnologias Midiáticas, com Defesa Pública em 30 de setembro de 2022, pela Banca constituída pelos seguintes professores:

ORIENTADOR

Prof. Dr. Eduardo Martins Morgado

Instituição: FAAC- Faculdade de Arquitetura, Arte e Comunicação

Presidente da Banca

Prof. Dr.^a. Ketilin Mayra Pedro

Instituição: UFSCAR- Universidade Federal de São Carlos

Prof. Dr. Myryan Lucia Ruiz Castilho

Instituição: UNIMAR- Universidade de Marília

Agradecimentos

Agradeço primeiramente a Deus pela sua infinita misericórdia. Pela sua presença em minha vida e meu coração, me impulsionando com força, fé, foco e sabedoria.

A minha família, principalmente minha amada mãe, Maria de Lourdes, que, mesmo em seus momentos difíceis, nunca deixou de ser a fortaleza de nosso lar. Você vem lutando e clamando por uma chance com todas suas forças, por isso minha eterna gratidão.

A meu marido, Edriano, que sempre me impulsiona e me recorda de todo meu percurso e de tudo que sou capaz com simples palavras de afeto e amor.

Aos meus dois amores, Lucas e Júlia, que são pedaços do meu coração que bate fora do meu peito. Vocês são meus bens mais preciosos, me fazem lembrar que tudo é possível. Aproveito para pedir desculpas pelas ausências necessárias, para que tudo isto acontecesse.

Aos meus irmãos, Giselda, Luciano e Edione, pelos momentos compartilhados e por fazerem parte de quem sou hoje, por trazerem minhas memórias nas suas.

A minha querida e amada sobrinha, Beatriz, que trouxe a nossas vidas o início de uma nova fase: a dos netos, sobrinhos e afilhados.

A minha sogra e meu sogro, que são minha rede de apoio, que nunca dizem não, que cuidaram quando não pude cuidar.

A meu orientador, Eduardo Morgado, pela oportunidade e confiança de sempre.

A meu coorientador, Wilson, pela amizade, correções e motivação, tão essenciais para que tudo desse certo.

A meus amigos que fizeram parte de todo o meu processo, não só nestes últimos anos, mas em toda minha vida. Não vou citar nomes para não me esquecer de ninguém, mas tenho certeza de que cada um sabe de sua importância.

A meus professores, que muito contribuíram em minha formação.

A meus alunos que fizeram parte de minha pesquisa. Vocês são responsáveis por esta conquista.

Às escolas em que lecionei e em que deixei um pouco de mim, levando muito de cada uma delas.

A todos que, direta ou indiretamente, fizeram parte desta caminhada.

*“É ótimo celebrar o sucesso, mas mais importante ainda é
assimilar as lições trazidas pelos erros que cometemos...”*
(Bill Gates)

RESUMO

CAMPANA, Samanta Bueno de Camargo. **Pensamento computacional na resolução de problemas na matemática**: Uma proposta de aplicação metodológica integrada na Educação Básica. 2022. 139 f. Tese (Doutorado em Mídia e Tecnologia) – Faculdade de Arquitetura, Artes, Comunicação e Design, Universidade Estadual Paulista “Júlio de Mesquita Filho”, Bauru, 2022.

O conhecimento eficaz é um fator importante para o desenvolvimento do aluno em todas as esferas de sua trajetória escolar e profissional. Portanto, muitos pesquisadores em educação matemática vêm buscando diferentes maneiras de melhorar a qualidade do ensino de matemática. O objetivo desta tese é investigar como os elementos do Pensamento Computacional são operados pelos alunos na resolução de problemas matemáticos. Considerando o objetivo geral, esta pesquisa tem como estratégia os seguintes objetivos específicos: Descrever o processo da resolução de problemas até chegar no algoritmo, identificar os elementos do pensamento algorítmico na resolução de problemas matemáticos e identificar como se dá o uso das habilidades de decomposição e algoritmo no processo da resolução de problemas. O ensino de matemática frente à resolução de problemas deve ser informal, intuitivo e conceitual, baseado em etapas que garantam um processo de desenvolvimento lógico, a fim de melhorar a qualidade da aprendizagem e desenvolver conceitos que ajudarão o indivíduo a solucionar diferentes problemas que exijam graus de dificuldade distintos. No primeiro momento, as habilidades do pensamento computacional foram desenvolvidas com o uso de atividades desplugadas. Em seguida, com os mesmos alunos do 9º ano do Ensino Fundamental, foi realizado um minicurso para que fosse possível a realização das atividades práticas. As atividades desenvolvidas foram elaboradas com o apoio de computadores e do ambiente de programação *Processing*. Em ambos os processos, foram verificadas habilidades do pensamento computacional na resolução de problemas matemáticos. Os dados da pesquisa foram coletados por meio de formulários, um minicurso e resoluções de atividades pertinentes aos objetivos da presente pesquisa.

Palavras-Chave: Pensamento Computacional; Resolução de Problemas; Matemática, Decomposição; Pensamento Algorítmico.

ABSTRACT

CAMPANA, Samanta Bueno de Camargo. **Computational thinking in problem solving in mathematics:** A proposal for an integrated methodological application in Basic Education. 2022. 139 f. Thesis (Doctorate in Media and Technology) – Faculty of Architecture, Arts, Communication and Design, São Paulo State University “Júlio de Mesquita Filho”, Bauru, 2022

Effective knowledge is an important factor for student development in all scope of school and professional trajectory. So many reserchers in mathematics education have been looking for diferent ways to improve the quality of mathematics education. The main of this thesis is investigate the entire process of student envolviment to solve a problem using computational thinking skills. Considering the general objective, this research has the following specific objectives as a strategy: To describe the problem solving process until arriving at the algorithm, to identify the elements of algorithmic thinking in the resolution of mathematical problems and to identify how the decomposition and decomposition skills are used. algorithm in the problem solving process. The teaching of mathematics in relation to solving problem must be informal, intuitive and conceptual, based on steps that assure a logical development process, in order to improve the quality of learning and concepts develop that will help the person to solve diferent problems that require degrees of diferent difficulties. In the first moment, computational thinking skills were developed using unplugged activities. Then, with the same students from the 9th grade of Elementary School, a short course was held to make it possible to carry out the practical activities. The developed activities were elaborated with the support of computers and the Processing programming environment. In both process, computational thinking skills were verified in mathematical problems solving. Research data were collected through forms, mini course and resolutions of activities relevant to the objectives of this research.

Keywords: Computational Thinking; Problem solving; Mathematics, Decomposition; Algorithmic Thinking.

Lista de figuras

Figura 1 - Média de matemática - Brasil x OCDE - avaliação do PISA 2019	33
Figura 2 - Eixos da abordagem didática da computação na Educação Básica.....	38
Figura 3 - Estágios para o PC	41
Figura 4 - Trajetória completa da decomposição	46
Figura 5 - Hexágono	48
Figura 6 - Decompondo: soma utilizando o padrão de Gauss.....	50
Figura 7 - Decompondo: soma por meio do padrão de Gauss	51
Figura 8 - Sequência triangular	52
Figura 9 - Sequência quadrangular.....	52
Figura 10 - Fluxo de informações em um algoritmo	54
Figura 11 - Algoritmo da fórmula de Gauss descrito em português	55
Figura 12 - Problema 1: resolução desplugada.....	57
Figura 13 - Código-fonte em <i>Processing</i>	58
Figura 14 - Habilidades do PC	59
Figura 15 - Eixos do PC articulados pela proposta da SBC	64
Figura 16 - Site oficial do Processing.....	66
Figura 17 - Ambiente <i>desktop</i> – <i>Processing</i>	70
Figura 18 - Código-fonte na linguagem <i>Processing</i>	71
Figura 19 - Exemplo de aplicação no <i>Processing</i>	71
Figura 20 - Entrada de dados na calculadora	73
Figura 21 - Exibição dos dados na calculadora.....	73
Figura 22 - Entrada dos dados no programa.....	74
Figura 23 - Exibição dos dados no programa.....	74
Figura 24 - Montando estampas de roupas - desenvolvimento de algoritmo.....	77
Figura 25 - Exemplo de códigos.....	78
Figura 26 - Jogo do estacionamento algorítmico – desafio 1	78
Figura 27 - Jogo do estacionamento algorítmico – D	79
Figura 28 - Modelo do tabuleiro do jogo com obstáculos (Haathi Mera Saathi).....	80
Figura 29 - Exemplos de cartões de comando do jogo (Haathi Mera Saathi).....	80
Figura 30 - Tabela de programação.....	81
Figura 31 - Estruturação da pesquisa	83
Figura 32 - Relação entre as variáveis do estudo	84

Figura 33 - Estágios para aplicação do PC em matemática	86
Figura 34 - Aplicação do minicurso	90
Figura 35 - Tabuleiro - torre de Hanói	97
Figura 36 - Resolução da atividade pelo participante 1	98
Figura 37 - Resolução da atividade pelo participante 4	99
Figura 38 - Resolução da atividade pelo participante 5	100
Figura 39 - Resolução da atividade pelo participante 6	101
Figura 40 - Participante 1: atividade 1	108
Figura 41 - Participante 1: atividade 3	108
Figura 42 - Participante 2: atividade 1	109
Figura 43 - Participante 2: atividade 3	110
Figura 44 - Participante 3: atividade 1	111
Figura 45 - Participante 3: atividade 3	112
Figura 46 - Participante 4: atividade 1	112
Figura 47 - Participante 4: atividade 3	113
Figura 48 - Participante 5: atividade 1	114
Figura 49 - Participante 5: atividade 3	115
Figura 50 - Participante 6: atividade 1	116
Figura 51 - Participante 6: atividade 3	117
Figura 52 - Participante 7: atividade 1	117
Figura 53 - Participante 7: atividade 3	119
Figura 54 - Participante 8: atividade 1	120
Figura 55 - Participante 8: atividade 3	121
Figura 56 - Etapas de resolução do primeiro grupo	122
Figura 57 - Etapas de resolução do segundo grupo	122
Figura 58 - Processo de resolução	125

Lista de quadros

Quadro 1 - Definição de autores acerca de PC	39
Quadro 2 - Problema envolvendo o conceito de decomposição.....	47
Quadro 3 - Decomposição do problema	47
Quadro 4 - Construção do hexágono	47
Quadro 5 - Decomposição do hexágono	48
Quadro 6 - Soma numérica.....	49
Quadro 7 - Soma de números	49
Quadro 8 - Soma em pares	54
Quadro 9 - Soma em pares com aplicação	54
Quadro 10 - Classificação dos participantes na primeira etapa das atividades	85
Quadro 11- Estrutura do minicurso	87
Quadro 12 - Cronograma do minicurso.....	89
Quadro 13 - Atividades desenvolvidas no Processing	90
Quadro 14 - Gosta de matemática/motivo/dificuldades ou facilidades.....	93
Quadro 15 - Integração de habilidades do PC na RP	94
Quadro 16 - Atividade desplugada - torre de Hanói.....	96
Quadro 17 - Atividade 1 – Processing – calculadora simples	103
Quadro 18 - Atividade 2 – Processing – verificação do mês do ano.....	104
Quadro 19 - Atividade 3 – Processing – soma dos números de 0 a 10	105
Quadro 20 - Atividade 4 – Processing – Tabuada.....	106
Quadro 21 - Atividade desplugada e plugada.....	123

Lista de tabelas

Tabela 1 - Artigos publicados sobre o uso do PC na educação24

Tabela 2 - Nível de rendimento em prova de larga escala, proficiência em matemática31

Sumário

CAPÍTULO 1 - PANORAMA E PROBLEMÁTICA	16
1.1 <i>Trabalhos correlatos</i>	19
1.2 <i>Objetivos</i>	23
1.2.1 <i>Objetivo geral</i>	23
1.2.2 <i>Objetivos específicos</i>	23
1.3 <i>Justificativa e motivação</i>	24
CAPÍTULO 2 – FUNDAMENTAÇÃO TEÓRICA.....	26
2.1 <i>Conceitos fundamentais e instrumental de aplicação</i>	26
2.2 <i>A resolução de problemas como metodologia e prática</i>	27
2.3 <i>RP e a sistematização do ensino de matemática na Educação Básica</i>	30
2.4 <i>PC, definição e aplicação</i>	33
2.4.1 <i>Sintetizando conceitos e definições do PC</i>	39
2.5 <i>Habilidades do PC</i>	41
2.5.1 <i>Decomposição</i>	41
2.5.2 <i>Decomposição na matemática</i>	43
2.5.3 <i>Decomposição no PC</i>	43
2.5.4 <i>Reconhecer padrões</i>	50
2.5.5 <i>Abstração</i>	53
2.5.6 <i>Algoritmo</i>	54
2.6 <i>Educação matemática e PC</i>	59
2.7 <i>O ambiente de programação do Processing</i>	65
2.7.1 <i>Breve resumo das origens do Processing</i>	65
2.7.2 <i>O ambiente Processing, derivados e evoluções</i>	66
2.7.3 <i>Bases do ambiente de programação do Processing</i>	67
2.7.4 <i>Ambiente desktop do Processing e suas funcionalidades</i>	69
2.7.5 <i>Utilizando o Processing na resolução de problemas matemáticos</i>	72
2.7.6 <i>Exemplos de utilização do Processing em conteúdos matemáticos</i>	72
2.8 <i>Atividades desplugadas</i>	75
CAPÍTULO 3 – PROCEDIMENTOS METODOLÓGICOS DA PESQUISA.....	82
3.1 <i>Delineamento da pesquisa</i>	82
3.2 <i>Metodologia de trabalho</i>	84

3.3 Participantes e local da pesquisa.....	84
3.4 Organização da coleta de dados.....	86
3.5 Desenvolvimento do minicurso	88
CAPÍTULO 4 – ANÁLISE E DISCUSSÃO.....	91
4.1 Perspectiva geral.....	91
4.1.1 Etapas desenvolvidas	92
4.2 Resultado da atividade desplugada	96
4.3 Processing, aplicação e resultados.....	101
4.3.1 Análise das atividades com o Processing	107
4.4 Análise das resoluções	107
4.5 Resultados da decomposição e algoritmo	121
CAPÍTULO 5 – CONSIDERAÇÕES FINAIS.....	126
REFERÊNCIAS	130
APÊNDICE 1 - Iniciativas nacionais e internacionais sobre PC	136

CAPÍTULO 1 - PANORAMA E PROBLEMÁTICA

De maneira geral, as ciências exatas, mais especificamente a matemática, emergem na história da humanidade a partir de necessidades muito primitivas e simples, tais como a contagem do rebanho, dos dias de trabalho e das estações, até as mais complexas demandas, como medições para construção e engenharia, além de abstrações e debates lógicos. Os exemplos de tais necessidades são inúmeros, bem como as soluções articuladas pelos mais diversos e numerosos pensadores. Esse é o caso de Euclides, na obra *Os Elementos*, cuja teoria elenca tanto uma infinidade de números primos quanto estudos geométricos, com os itens que dão título à obra definidos e articulados entre si, tais como ponto, linha, ângulo, segmento, proporção e superfície, os quais permanecem, em sua essência, úteis à resolução de problemas do cotidiano até os dias de hoje.

Essa necessidade de conseguir soluções para os problemas corriqueiros se constituiu como uma das bases das reflexões do filósofo Sócrates, que, já em meados do século IV a.C., afirmava que o ser humano possui o conhecimento fundamental para resolver problemas, sendo essa capacidade algo nato ao indivíduo, consistindo em um mero exercício de recordação. Em seu conhecido *Diálogo Socrático*, o filósofo ilustra o pensamento com a apresentação e uso de algumas técnicas próprias, por meio de um monólogo indagativo em que, ao apresentar uma questão, desenvolve o percurso solucionador, de maneira a estimular a criação de uma resposta, a própria resolução do problema, na medida em que também elabora o pensamento autônomo.

Dentro desse cenário, se pensarmos nas vivências pedagógicas discentes que se efetivam com a resolução de problemas na matemática, temos que dada a abstração necessária para trabalhar com problemas que não são concretos (embora possam ser extraídos do cotidiano), muitas das questões apresentadas nas aulas de matemática são resolvidas sem que se tenha a certeza do que está sendo feito, ou sequer se a matemática está relacionada com aquilo, o que gera uma série de questões essenciais ao processo de ensino-aprendizagem¹. Papert afirma que é necessário favorecer a construção de conhecimento sempre fazendo uma relação e contextualização entre o concreto e abstrato. Papert (1986, p. 10).

Sendo assim, cabe perguntar: quais conhecimentos matemáticos o estudante utiliza no processo de resolução de problemas? O que podemos observar nessa resolução? Ou ainda: qual seria a melhor maneira de resolver um problema no âmbito do ensino da matemática,

¹ Esta afirmação se justifica pela experiência docente da pesquisadora na área de ensino de matemática, sua vivência profissional e carreira na área, trabalhando com essas questões ao longo de toda a sua carreira.

independentemente da sua complexidade? Quais conhecimentos matemáticos deveriam estar presentes nesse processo?

A maior parte dos alunos segue fórmulas ou regras dadas pelo professor sem ao menos entender o processo, ou ainda sem conseguir perceber erros dentro de tal resolução, cheia de números ou sequências aleatórias, muito distante de uma aplicação prática e efetiva.

Ou seja, é preciso entender que não bastam a necessidade de resolver o problema e a disposição de ferramentas, fórmulas e conhecimento de operacionalização básica de conceitos abstratos: para que a resolução seja assertiva e contextualizada na realidade do indivíduo e assim se concretize e gere efetivo conhecimento, é necessário desenvolver uma série de habilidades. Pois dados do Saresp, Enem e Pisa e demais avaliações de larga escala nos mostram o fracasso escolar na área da matemática.

Dessa forma, esta pesquisa trabalha com a necessidade de instrumentalizar a metodologia da resolução de problemas no ensino da matemática e, para tanto, visa apresentar o Pensamento Computacional (PC) como um instrumento possível para o desenvolvimento das habilidades necessárias ao aluno na aplicação da resolução de problemas que envolvem matemática. Com base nesse pressuposto, no presente trabalho, por meio da aplicação de atividades e de outras estruturas correlacionadas, intentamos mostrar o refinamento que se obtém no decorrer do processo de resolver um problema utilizando o pensamento computacional.

Para isso, a presente tese realiza uma revisão de literatura, elencando autores e trabalhos de amplo reconhecimento acerca do conceito de PC, com a intenção de demonstrar que tais conceitos de utilização conjunta da matemática e da computação no âmbito da educação coadunam em uma possível e satisfatória aplicação do pensamento computacional, propiciando consistência didática e pedagógica no planejamento dos estudos matemáticos empreendidos sob a metodologia da resolução de problemas. Assim, utilizamos critérios e parâmetros que englobam os processos de:

- decomposição;
- reconhecimento de padrões;
- abstração; e
- algoritmo.

Dessa forma, e com o intuito de trabalhar uma introdução a essas questões no âmbito desta tese, neste primeiro capítulo, serão expostos os conceitos acerca do pensamento computacional relacionado à resolução de problemas matemáticos, levando em conta diferentes peculiaridades das habilidades do pensamento computacional.

Assim, este capítulo tem por objetivo apresentar um estudo pormenorizado acerca de duas habilidades, o pensamento algoritmo e a decomposição. A partir dessas duas habilidades, haverá um aprofundamento acerca do que são atividades desplugadas e de aplicações com o ambiente de aprendizagem *Processing*.

Pesquisadores como Blikstein e Wilensky (2009); diSessa (2001) e Wing (2006) apontam que a utilização do PC vem auxiliando de maneira positiva o desempenho no processo de ensino-aprendizagem nos mais diversos níveis da educação.

Os princípios do PC, por muito tempo, eram reconhecidos somente pelos estudiosos de ciência da computação, mas atualmente os olhares de diferentes áreas do conhecimento têm se voltado para os benefícios dessas práticas, como instrumento de otimização e melhoria na aprendizagem escolar, em especial no ensino de matemática. São variados os exemplos disso: aqui citamos um dos primeiros (e mais importantes, especialmente por seu pioneirismo) *softwares* educacionais a utilizar linguagem de programação, o *Logo*², cujo objetivo é a utilização de procedimentos computacionais que facilitem a aprendizagem de certos conceitos modelados e computacionais, como abstração ou construção de algoritmos (PAPERT, 1980) *apud* RODRIGUES; ANDRADE; GUERRERO; CAMPOS, 2015). É muito utilizado até os dias de hoje para o desenvolvimento do aprendizado de matemática.

Sendo assim, uma vez que o presente trabalho visa ao desenvolvimento de habilidades educacionais, buscamos oferecer, por meio desta pesquisa e de seus desdobramentos e resultados, práticas de ensino com mais qualidade, que objetivem a equidade no ambiente escolar. Também intencionamos que, com o desenvolvimento das habilidades correlacionadas às áreas da matemática e da computação, tais práticas estejam entrelaçadas com os objetivos e diretrizes da ONU, na Agenda 2030, especialmente com o Objetivo de Desenvolvimento Sustentável (ODS), viabilizando o instrumental para a educação de qualidade para todos e o desenvolvimento de habilidades específicas, independente da classe social ou da natureza da escola (se pública ou particular), para que haja um desenvolvimento pleno dos conhecimentos e capacidades dos alunos, conforme prevê este objetivo.

Ao longo deste trabalho, no tocante à resolução de problemas simples ou complexos, o objetivo é que a combinação da resolução de problemas com as ferramentas e os métodos do PC assegurem que a Educação Básica consiga oferecer atividades que desenvolvam competências específicas que serão utilizadas ao longo dos anos e da vida do indivíduo como um todo, perdurando e perpassando a esfera educacional para o cotidiano.

² A linguagem de programação *Logo* é uma ferramenta utilizada para o desenvolvimento de habilidades na programação de computadores destinada para crianças e jovens.

Dessa forma, o objetivo geral deste trabalho é, investigar como os elementos do PC são operados pelos alunos na resolução de problemas matemáticos. Proceder à coleta de dados por meio de atividade escolares com metodologia e desenvolvimento próprios, que demonstrem a aplicação e o potencial de expansão não só do PC, mas de outros conhecimentos próprios da computação no contexto do ensino da matemática.

É fato que o uso do PC está cada vez mais presente, atualmente nas grades curriculares das universidades, como o caso da UNIVESP, por meio de aulas próprias e calcadas na robótica e na programação, em vez de integrado com o currículo de maneira interdisciplinar, e, com essa discussão, elencar os benefícios dessa integração, em especial com a matemática, tanto em estrutura lógica quanto no que se refere à capacidade de modelar e investigar relações lógico-matemáticas. Por fim, objetivamos descrever o processo da resolução de problemas até chegar no algoritmo, identificar os elementos do pensamento algorítmico na resolução de problemas matemáticos, identificar como se dá o uso das habilidades de decomposição e algoritmo no processo da resolução de problemas.

Sistematizar a relação entre os conteúdos e os conceitos matemáticos e computacionais, combinando e utilizando conjuntamente duas habilidades do PC, a saber, a decomposição e o algoritmo, o que deve ser iniciado com atividades *offline* (fora da rede, fora do computador), para, por fim, chegar a atividades *online*, conectadas com a utilização do ambiente de programação *Processing*.

Tanto a questão-problema desta pesquisa quanto sua aplicação justificam-se na medida em que hoje trabalhamos com uma perspectiva pervasiva da computação e das novas tecnologias, que permeiam e se expandem por todas as áreas do conhecimento humano, seja nas ciências exatas, biológicas ou humanas. Ou seja, quando propomos uma união entre a computação e essas áreas de conhecimento no âmbito da interdisciplinaridade, visamos a trabalhar com a perspectiva dos benefícios ao desenvolvimento humano, uma vez que os princípios e os modelos da própria computação são pontes para resolver problemas.

Justifica-se, portanto, o esforço em oferecer, com esta pesquisa, uma proposição de base teórica e de modelo prático para a abordagem de todo o processo dessa relação, que pode ser bastante benéfica e positiva no ensino de matemática na Educação Básica, fundamental e médio.

1.1 Trabalhos correlatos

A seguir estão alguns trabalhos correlatos à pesquisa que contribuíram para compreensão, de forma mais abrangente, do problema e dos objetivos desta pesquisa.

A relação entre o PC e o ensino de matemática estabelece intersecção identificada na pesquisa de Barcelos e Silveira (2012), em que os autores exploram os documentos relativos aos Parâmetros Curriculares Nacionais em seu contexto complementar (BRASIL, 2002) e destacam que as possibilidades de uso em conjunto do PC e da matemática devem estar bem definidas.

Os autores definem no estudo três momentos em que a correlação dos temas é mais eficiente. São estes:

- Articulação de símbolos e códigos: aprender algoritmos pode permitir ao estudante criar e testar hipóteses desenvolvendo o conhecimento matemático por meio desse aprendizado.
- Estabelecimento de relações e identificação de regularidades: aqui os autores defendem que a identificação de regulares ou sequências numéricas (como Fibonacci, por exemplo) seria facilitada se os estudantes aprendessem os fundamentos do PC.
- Modelos explicativos e representativos: neste tópico Barcelos e Silveira (2012) exploram os conceitos probabilísticos mais complexos que podem ser desenvolvidos na relação entre matemática e PC, indo além do letramento computacional e utilizando *softwares* mais poderosos.

Barcelos e Silveira (2012) defendem que essas três competências podem ser testadas durante o uso do PC e da matemática em aula, dessa forma propiciando mais possibilidades de os docentes compreenderem, com maior efetividade, os benefícios que o PC e sua disseminação podem propiciar ao contexto escolar.

Outro estudo que discute a relação entre o PC e o aprendizado de matemática é Rich *et al.* (2019). Após destacarem que os professores são o primeiro elo de relação entre o PC e matemática, os autores buscam alternativas para adaptar os fundamentos do PC para os seus estudantes, tendo em vista que esse conhecimento não consta obrigatoriamente na grade curricular.

Rich *et al.* (2012) destacaram a relação entre ciência e PC por meio de um seminário de atividades conduzidas com os docentes. É importante também citar que o estudo conduzido busca as melhores formas de utilizar a bagagem de conhecimento do professor em sua atividade diária para que as atividades com relação ao PC sejam mais efetivas.

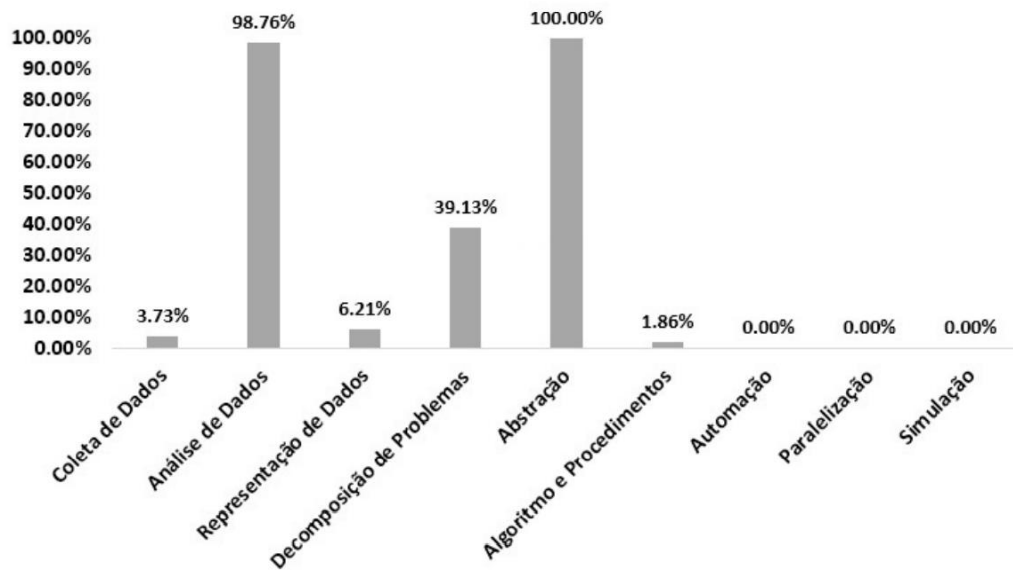
Por fim, os autores dão enfoque à importante relação entre PC e matemática, defendendo que a educação deve estar progressivamente mais conectada ao conhecimento de ciência da computação, dado estarmos cada vez mais imersos em um mundo muito conectado à internet.

O estudo de Costa, Sampaio e Guerreiro (2016) é mais uma pesquisa que também relaciona PC e matemática como uma forma de trabalhar, junto aos alunos, uma competência de resolução de problemas mais bem apurada. Nesse trabalho, os autores apresentam a estudantes de 8º e 9º ano do ensino fundamental competências relacionadas ao uso do PC.

Os dados desse mesmo estudo levam à conclusão de que a decomposição é a competência do PC mais bem conectada com a matemática. Em seguida, expõem que outras habilidades, como análise de dados e abstração, também estabelecem relação com a matemática, mas em um nível inferior ao constatado pela decomposição.

Um outro artigo que faz análise empírica da relação entre PC e ensino de matemática é Mestre *et al.* (2015). Os resultados do estudo concluem que, das nove competências do PC avaliadas — coleta de dados, análise de dados, representação de dados, decomposição de problemas, abstração, algoritmo e procedimentos, automação, parametrização e simulação —, seis têm relação com a avaliação do PISA 2012 (*Programme for International Student Assessment*) (LORENCEAU; MAREC; MOSTAFA, 2019): abstração, análise de dados, decomposição de problemas, representação de dados, coleta de dados e algoritmos e procedimentos. Tal avaliação é de cunho internacional e permite constatar a qualidade educacional dos muitos países que submetem seus estudantes a essa prova. O PISA também serve de parâmetro para se compreender o que os melhores sistemas educacionais do mundo fazem para atingir resultados de excelência.

Para Mestre *et al.* (2015), as competências das questões de matemática do PISA estão mais fortemente relacionadas a conceitos de PC como abstração, análise de dados, decomposição de problemas e representação. Ainda segundo os autores, os dados permitem extrair indícios de que o ensino de PC na Educação Básica pode colaborar para o processo ensino-aprendizagem voltado à resolução de problemas. O Gráfico 1 apresenta os resultados dos autores:

Gráfico 1 - Distribuição dos conceitos do PC identificados nas questões do PISA

Fonte: Mestre et al. (2015).

Outro estudo, realizado por França *et al.* (2016), destacaram a interdisciplinaridade referente ao PC como uma forma de desenvolver competências que transcendem apenas o letramento digital, como aprendizado de educação matemática, jogos digitais e robótica. Os autores ainda ressaltam que o uso da tecnologia atualmente se aplica a um grande número de atividades profissionais, como à área da saúde, por exemplo.

Em estudo de França e Tedesco (2016), discute-se que os currículos escolares ainda não se adaptaram ao ensino de PC. Assim, os autores exploram a ideia de, por meio do desenvolvimento de jogos digitais, o ensino de PC pode contribuir efetivamente para o aprendizado como um todo.

Os autores buscam traçar um panorama nacional em que pesquisadores de diversas regiões do país se envolvem na pesquisa de PC na Educação Básica. Também comentam o esforço dos docentes para que entendam os conceitos de PC e como isso se reflete tanto no currículo dos alunos quanto na formação dos professores.

Blikstein (2008) ilustra, de forma didática, como o PC pode paradoxalmente nos tornar dependentes do uso de dispositivos eletrônicos e desenvolver autonomia no aprendizado, no que se refere ao uso de tecnologias em sala de aula:

Precisamos urgentemente redirecionar nossos esforços e recursos para ensinar nossas crianças a compreender esse interessante paradoxo: o pensamento computacional nos torna cada vez mais dependente e, ao mesmo tempo, diferentes dos computadores. Entender como podemos ser mais produtivos e criativos –

mesmo sendo mais dependentes – é o maior desafio dos educadores que querem repensar a tecnologia na sala de aula. (BLIKSTEIN, 2008, p. 6).

Essa perspectiva ressalta a aderência da intersecção entre PC e ensino de matemática discutida pelos artigos aqui expostos. Como citado nesses mesmos trabalhos, o PC vai além, no sentido de a capacidade de resolução de problemas que proporciona se estender a modelos matemáticos que podem ser utilizados por outras áreas. Sendo assim, a integração do PC à Educação Básica ultrapassa a matemática, sendo uma proposta enxergada pelos autores como interdisciplinar.

1.2 Objetivos

O PC tende a ser visto como o objetivo em si, em vez de integrado com o currículo. Porém, visamos aqui a relacionar as habilidades do PC e da matemática, tanto em estrutura lógica quanto na capacidade de modelar e investigar relações matemáticas. Intencionamos analisar como o aluno desenvolve os problemas, observando características do PC e como o uso das habilidades de decomposição e de abstração bem-desenvolvidas pode auxiliar na resolução de problemas simples e complexos.

1.2.1 *Objetivo geral*

Investigar como os elementos do PC são operados pelos alunos na resolução de problemas matemáticos.

1.2.2 *Objetivos específicos*

Considerando o objetivo geral, esta pesquisa tem como estratégia os seguintes objetivos específicos:

- Descrever o processo da resolução de problemas até chegar no algoritmo.
- Identificar os elementos do pensamento algoritmo na resolução de problemas matemáticos.
- Identificar como se dá o uso das habilidades de decomposição e algoritmo no processo da resolução de problemas.

1.3 Justificativa e motivação

A presente pesquisa se debruça sobre conceitos e habilidades relacionadas ao PC. Wing (2006) afirma que tais conhecimentos são fundamentais para o desenvolvimento da sociedade do século XXI, tornando-se um tema de interesse para a pesquisa e para inovações e intervenções nas áreas de ciências, tecnologia, engenharia e matemática (GROVER; PEA, 2013). Nesse sentido, esta pesquisa busca estabelecer uma relação entre a resolução de problemas na disciplina de matemática e o uso das habilidades do PC.

O uso de computadores na resolução de problemas é considerado como um recurso em potencial, que possibilita resoluções de forma criativa e bem estruturadas, sendo o processo de pensamento um ponto importante para representar soluções eficazes realizadas pelo processamento da informação (WING, 2010).

Essas ferramentas tecnológicas facilitam e moldam a maneira como as pessoas pensam e desenvolvem conceitos matemáticos. É a partir desse pressuposto que a presente pesquisa vem propor uma investigação de como esse processo de resolver problemas utilizando habilidades do PC acontece.

Após as publicações de Jannette Wing sobre o PC, foram relatadas diversas experiências de seu uso na educação pelo Brasil, por exemplo, o trabalho de Araújo, Andrade e Guerrero (2016). Eles fazem um levantamento nacional sobre a abordagem do PC pelas regiões do país, especificamente por meio de artigos científicos. A Tabela 1 apresenta a quantidade de projetos por região em ordem decrescente.

Tabela 1 - Artigos publicados sobre o uso do PC na educação

Região	Trabalhos
Nordeste	13
Sul	6
Sudeste	2
Centro-Oeste	1
Norte	0

Fonte: adaptada de Araújo, Andrade e Guerrero (2016).

Os autores ainda classificam os assuntos mais abordados relacionados ao uso do PC. Nessa classificação destaca-se o *Scratch*, desenvolvido pelo MIT, com ocorrência mais frequente. As atividades desplugadas ocupam a segunda posição em quantidade de experiências desenvolvidas com os estudantes.

Dessa forma, intentamos utilizar o referencial teórico existente para propor uma investigação a partir de um ambiente de programação, o *Processing*, desenvolvido pelo MIT. Há cada vez mais usuários dessa nova aplicação, uma vez que não é necessária a compra de licenças para seu uso, além de atualmente já possuir existir versão *Web*, de modo que basta um navegador conectado à internet para a sua utilização.

Assim, este trabalho se dá em duas etapas. A princípio, apresentamos uma atividade inicial com resolução de problemas desplugado, em que não é necessário o uso de dispositivos eletrônicos para a sua resolução, ressaltando assim a associação entre conceitos de PC e matemática não necessariamente relacionada somente a alfabetização digital (em que o estudante apenas aprende a operar os *softwares* de um computador, por exemplo). Posteriormente, expomos o desenvolvimento de resoluções de problemas envolvendo o ambiente de aprendizagem *Processing*.

O aluno aprende tanto por meio das atividades desplugadas quanto com o apoio do *Processing*, em dispositivos eletrônicos como *laptops* ou *desktops*. Isso possibilita autonomia ao estudante, na medida em que ele consegue reutilizar estratégias para resolver problemas simples e complexos. Por isso, analisamos como esse aluno faz uso da decomposição e do pensamento algoritmo em seu processo de resolução, podendo assim verificar como esse processo influencia o desenvolvimento lógico e contribui para que o estudante consiga sofisticar seu entendimento e seu aprendizado.

CAPÍTULO 2 – FUNDAMENTAÇÃO TEÓRICA

O presente capítulo perfaz o estudo sobre os conceitos fundamentais de resolução de problemas como metodologia no ensino de matemática no âmbito da Educação Básica. Abordamos diferentes temas relacionados ao PC, como sistematização do conceito, aplicação e habilidades. Colocamos a resolução de problemas como um desdobramento mais específico do estudo do PC, assim como descrevemos atividades desplugadas e a aplicação do ambiente de programação *Processing*.

2.1 Conceitos fundamentais e instrumental de aplicação

Nos últimos anos, com o refinamento dos princípios e das normas referentes à matemática escolar, tornou-se claro que o ensino e a aprendizagem da matemática estão baseados em padrões e são mecanizados não só para o professor, mas também para o aluno. Como cita Romanatto (2012), para além da resolução de problemas, a matemática nos leva a pensar, assim como a descobrir padrões e conexões.

Entretanto, a aquisição da matemática como conhecimento por meio da resolução de problemas tem sido considerada uma das principais formas de inovar na compreensão e no auxílio para o desenvolvimento do conhecimento lógico-matemático. Como ainda nos coloca Romanatto (2012), a resolução de problemas deve primar por mudanças que passem pela teoria e prática.

É certo que as crianças e os jovens estão imersos na resolução de problemas, na medida em que experimentam e intuitivamente contemplam como as coisas funcionam (por exemplo, o que o ambiente torna quente ou frio), e uma vez que uma série de manipulações e repetições são necessárias para aprender, quem sabe, como tocar um instrumento musical ou amarrar um cadarço. Se os professores e os responsáveis facilitarem esse tipo de experimentação com modelos, diagramas e, finalmente, números, parece que a linha tênue entre a resolução de problemas mecanizada por fórmulas prontas e fechadas em suas resoluções únicas desapareceria (ROMANATTO, 2012). Em outras palavras, a matemática ganharia fluência e flexibilidade e, ao mesmo tempo, automaticidade, com traços menos convencionais.

Porém, temos uma situação que é vivida por muitos professores na sala de aula: aparentemente sobrecarregados com longas jornadas de trabalho, extensas cargas horárias e cobranças de resultados para melhora dos índices das avaliações de larga escala, os docentes ainda optam pelos métodos mais mecanizados, pois a maioria prefere trabalhar dentro de uma

zona de conforto previsível durante suas aulas (CARVALHO; PENTEADO, 2001), com fórmulas e equações prontas, a fazer uso da proposição ao aluno do pensamento criativo, oportunizando a resolução de problemas e ensinando a pensar. Uma breve observação, mesmo em pequena escala, acaba por mostrar que é certo que esse modelo não vem funcionando e que o método de resolução de problemas se mostra mais próximo à realidade de aplicação e do uso do estudante.

Ao longo das últimas décadas, vários autores veem a resolução de problemas como uma possibilidade para a melhora da compreensão dos conteúdos relacionados à matemática. Na década de 1970, construiu-se a ideia da resolução de problemas como forma consciente para se atingir um determinado objetivo (POLYA, 2006). Já na década de 1980, são incluídas as narrativas dos jogos e dos desafios que estimulam a resolução de problemas de forma mais abrangente (THOMPSON, 1989). Já nos anos 1990 e no início dos anos 2000, a ideia da resolução de problemas se consolida como uma forma de o estudante conseguir definir resoluções próprias e não apenas seguir um determinado itinerário para resolver problemas matemáticos (ONUCHIC, 1999; ONUCHIC; ALLEVATO, 2004; VAN DE WALLE, 2009).

2.2 A resolução de problemas como metodologia e prática

Embora muitas áreas utilizem a resolução de problemas como metodologia, no ensino de matemática, a conjunção e a aplicação dessa metodologia são notáveis na Educação Básica. Nesse sentido, é necessário que entendamos, em primeiro lugar, o que, para essa abordagem metodológica, define-se como problema. Dante (1989, p. 9) conceitua que “problema é qualquer situação que exija o pensar do indivíduo para solucioná-lo”, o que não se refere a problemas necessariamente matemáticos, mas a de qualquer natureza. O autor também considera que, apesar de os olhares docentes se voltarem à resolução de problemas, esse ainda é um conteúdo pouco assimilado pelos alunos, pois as abordagens, em sua maioria, não motivam os indivíduos a efetivamente desenvolver o pensamento autônomo e estratégico, bases de tal metodologia pedagógica. Há, assim, uma necessidade de estimular e envolver o aluno na resolução de problemas (doravante RP), para que, dessa forma, ela passe a ser produtiva em seu processo de ensino-aprendizagem.

Em sua tese de livre-docência, Dante (1989) aborda alguns pontos importantes da RP, além de algumas etapas para serem desenvolvidas nos estudantes durante os anos iniciais.

Fazer com que o aluno pense produtivamente; desenvolver o raciocínio do aluno; preparar o aluno para enfrentar situações novas; dar oportunidades aos alunos de se envolverem com suas aplicações; tornar as aulas de matemáticas mais interessantes e desafiadoras; equipar o aluno com estratégias e procedimentos que auxiliam na análise e na solução de situações onde se procura um ou mais elementos desconhecidos. (DANTE, 1989, p. 15-16).

Ainda que atualmente na RP muitos alunos façam uso de diversos algoritmos, nem por isso conseguem ou estão aptos a efetivamente resolver os problemas propostos com tais algoritmos. Dante (1989) acredita que isso acontece por confundirem o exercitar com o pensar, sendo necessário diferenciar as modalidades de exercício de RP, sendo a primeira categoria a do exercício de treino, que visa a treinar e aperfeiçoar uma técnica por meio da repetição de padrões, e a segunda categoria a do exercício que se configura e se define pela busca efetiva de uma solução, processo este em que não há um algoritmo definido que proporcione a sua definição, pois tudo gira em torno de pensar na estratégia da aplicação. Realizadas essas articulações, Dante acredita que, uma vez que o aluno tenha contato adequado com a RP, o processo passe a ser natural durante todo o ciclo de sua vida, em qualquer esfera, mesmo nos mais complexos âmbitos, como economia ou política.

Um dos pioneiros a abordar o tema em pesquisa, fomentando uma discussão sobre o assunto, é Polya (2006), que define a metodologia como “a atividade matemática mais próxima do cotidiano”. O matemático acredita que, quando se resolve um problema, uma saída é descoberta e, dessa forma, o processo consiste na busca por um caminho que proporcione a chegada, ainda que contornando obstáculos, a uma solução, mesmo que ela não se apresente de imediato. Dessa forma, para Polya (2006), o desenvolvimento da habilidade de resolver problemas deveria ser o foco principal das aulas de matemática.

A resolução de problemas é uma habilitação prática como, digamos, o é a natação. [...] Ao tentarmos resolver problemas, temos de observar e imitar o que fazem outras pessoas quando resolvem os seus e, por fim, aprendemos a resolver problemas, resolvendo-os. (PÓLYA, 2006, p. 4).

De modo a resolver problemas de maneira assertiva, o autor cria e apresenta um método para RP, conhecido como método Polya (2006) para RP de matemática, sendo dividido em quatro etapas:

- 1ª Etapa - compreender o problema.
- 2ª Etapa - elaborar um plano.
- 3ª Etapa - executar o plano.
- 4ª Etapa - fazer uma verificação.

O método de solução com estratégias heurísticas de Polya (2006) consiste em uma teoria de execução bem-aceita e utilizada por muitos estudiosos da área, e há inclusive quem faça uso dessa teoria mesmo sem conhecê-la a fundo, tamanha sua difusão. Ao abordar a primeira etapa, trabalhando a compreensão o problema, o autor afirma que é importante o aluno entender todo o enunciado, elaborando questionamentos que o ajudem na compreensão da resolução. Esses questionamentos devem estar relacionados ao problema propriamente dito, por exemplo:

- Quais os dados do problema?
- Qual é a incógnita?
- Qual a condição para o problema ser resolvido?
- Às condições são satisfatórias para encontrar as incógnitas?

No momento que o aluno tenta compreender o problema, no passo um, vai realizando a abstração dele. Nessa etapa a concentração e a atenção são voltadas para pontos importantes, de modo que os pequenos detalhes passam a ser irrelevantes, o que remete à condição de apurar características deixando de lado as particularidades em todo o processo ao se resolver o problema. Nessa etapa o estudante está realizando a abstração do problema.

Já na segunda etapa, faz-se necessário elaborar um plano para estabelecer uma relação concreta entre os dados que se obtêm do problema e a incógnita. Nesse momento, emerge a necessidade de buscar uma relação entre o que se está propondo a resolver e alguns correspondentes, já solucionados, por similaridade. O problema pode ser redigido de maneira diferente, para se tornar mais familiar, e há questionamentos que podem ajudar nesse processo de reconhecimento, tais como: “Você já viu esse dado anteriormente?”.

Nessa etapa, depois que compreendeu o problema, o aluno passa a configurar um plano, construir o algoritmo, definindo uma sequência de etapas finitas para sua resolução, focando o uso de diferentes estratégias para desenvolver sua resposta. Elas podem incluir a busca por padrões e a resolução de um problema equivalente, encontrar uma fórmula ou coordenada, ou mesmo resolver equações, sendo as variáveis o suporte para o armazenamento de informações de resolução.

O autor destaca a importância dessas primeiras etapas para uma boa compreensão do problema, e se, fortuitamente, elas não forem suficientes para a compreensão e a elaboração de uma estratégia, ainda se faz necessário observar o problema sob outra perspectiva, até que seja suficiente.

Na terceira etapa, é momento de trabalhar a execução do plano, aplicando o procedimento escolhido na etapa anterior, certificando-se de que esteja sendo realizado adequada e corretamente. Na quarta etapa, é necessário solucionar o problema, buscando a validação dos

resultados e conferindo todos os procedimentos realizados. Se ocorreu algum erro durante a resolução, reconsidera-se cada etapa executada para se obter o resultado, revisitando todo o caminho percorrido para obtenção da resolução, de modo a perceber as diferentes possibilidades existentes e possíveis.

De modo a estabelecer uma conclusão sobre a aplicabilidade da heurística de Polya (2006), temos que esse método direciona os alunos a resolverem problemas por meio de estratégias geradas por eles mesmos, partindo do princípio de que é necessário ler e entender o problema. Para isso, as estratégias podem fazer o aluno:

- Reescrever o problema de maneira mais simplificada.
- Retornar, sempre que necessário, para rever seus passos.
- Verificar cada etapa em busca de validação.

Estudar como os matemáticos pensam ou definem a RP nos dá embasamento para partirmos de estudos e *scores* já estudados, de modo a fortalecer nossa pesquisa, verificando a necessidade de se pensar em estratégias para melhoria na RP e poder analisar todas as nuances que ocorrem durante essa resolução. Sabemos que qualquer estratégia para RP exige direcionamento do educador, além de modelagem, estratégias e muita prática, para que o aluno consiga trocar problemas simples por mais complexos.

2.3 RP e a sistematização do ensino de matemática na Educação Básica

Se entendermos que a educação, de forma geral, consiste no conjunto de processos que objetiva o desenvolvimento potencial de saberes, de conhecimentos específicos e técnicos e de linguagens em cada indivíduo, ao mesmo tempo que possibilita a socialização e o aprendizado em comunidade para o crescimento cidadão, podemos concluir que a disciplina de matemática se insere nesse contexto, de maneira a propiciar o entendimento de questões práticas, técnicas, abstratas e lógicas imprescindíveis à autonomia do indivíduo em sociedade. No entanto, ela é, com frequência, percebida pelos alunos como sendo uma disciplina de difícil compreensão, cujos problemas de apreensão e compreensão vão emergindo conforme o aluno vai se distanciando da matemática mais concreta, nos anos iniciais, a um nível mais abstrato.

Sendo a matemática uma disciplina caracterizada pela necessidade de manter a precisão e a clareza tanto de seus processos quanto de seus resultados, de modo a se obter desempenho satisfatório, é necessário dar atenção aos pequenos detalhes, quer na parte processual, quer nas explicações de como realizar uma tarefa ou escrever um algoritmo. Todavia, ainda que haja preocupação e esforço de muitos profissionais da área da educação para que os alunos atinjam

seu potencial, a maioria deles apresenta problemas sérios nessa disciplina, condição perfeitamente comprovada por dados oficiais da educação brasileira.

No Brasil, o governo federal utiliza o Sistema de Avaliação da Educação Básica (SAEB) para avaliar os alunos no final de cada ciclo: nos 5º e 9º anos do Ensino Fundamental e no 3º ano do Ensino Médio. Essa avaliação acontece a cada dois anos e se dá pela composição das médias extraídas da Prova Brasil e pelo Índice de Desenvolvimento da Educação Básica (IDEB). No ano de 2017, o Ministério da Educação (MEC) utilizou as seguintes escalas com nível de proficiência: insuficiente, básico e adequado. Nesse mesmo ano, no resultado referente à disciplina de matemática, nota-se que três em cada dez alunos no Ensino Médio têm nível insuficiente. Dentro dessa mesma categoria, são 22,49% que se encontram no patamar mais baixo da avaliação, ou seja, com índice de proficiência zero, conforme tabela a seguir:

Tabela 2 - Nível de rendimento em prova de larga escala, proficiência em matemática

Níveis	Insuficiente% Entre (0 e 3)	Básico% (4 e 6)	Adequado% (7 e 9)
5º ano	33,12	51,35	15,52
9º ano	63,11	32,39	4,5
3º ano	71,67 <i>Destes 22,49 estão no nível zero</i>	23,81	4,52

Fonte: MEC (2017).

Observando os índices das avaliações dos anos anteriores a 2017, notamos ainda que o Ensino Fundamental vem apresentando piora gradativa e que, ao longo dos anos, ultrapassará o Ensino Médio em índices negativos, já que apenas 4,5% dos alunos se encontram no nível considerado adequado, e 63,11%, no insuficiente³.

No que se refere às habilidades matemáticas, tais exames de larga escala e abrangência dão ênfase à RP, de modo a verificar se o aluno consegue estabelecer conexão significativa com os algoritmos estudados dentro de situações cotidianas, procurando contextualizar e estabelecer relações no momento da resolução.

O conhecimento de Matemática no Saeb é demonstrado por meio da resolução de problemas. São consideradas capacidades como observação, estabelecimento de relações, comunicação (diferentes linguagens), argumentação e validação de processos, estimulando formas de raciocínio como intuição, indução, dedução e estimativa[...] quando os alunos têm situações desafiadoras e trabalham para desenvolver estratégias de resolução, o que não exclui totalmente a possibilidade

³ Os dados citados podem ser encontrados em:

https://download.inep.gov.br/publicacoes/institucionais/avaliacoes_e_exames_da_educacao_basica/relatorio_saeb_2017.pdf e os parâmetros em: <https://qedu.org.br/brasil/proficiencia>.

da proposição de alguns itens com o objetivo de avaliar se o aluno tem domínio de determinadas técnicas. (BRASIL, 2019a, p. 27).

Para que esses dados e índices avaliativos não continuem descendendo, é necessário compreender como a RP está sendo aplicada, onde se encontram as lacunas no desenvolvimento das atividades e quais estratégias podem ajudar a melhorar tais habilidades no desenvolvimento de todo o processo da RP. Nesse sentido, são necessárias políticas públicas e pesquisas que propiciem a melhoria do ensino de matemática, além de estabelecer boas práticas e procedimentos adequados para a abordagem dos conteúdos no momento do ensino, de modo a descaracterizar o pensamento obsoleto e errôneo de que aprender matemática nada mais é do que simplesmente memorizar fórmulas, seguir modelos e processos estáticos ou aplicar algoritmos de dígitos aleatórios.

[...] o professor que acredita que o aluno aprende Matemática por meio da memorização de fatos, regras ou princípios transmitidos pelo professor ou pela repetição exaustiva de exercícios, também terá uma prática diferenciada daquele que entende que o aluno aprende construindo os conceitos a partir de ações reflexivas sobre materiais e atividades, ou a partir de situações-problema e problematizações do saber matemático. (FIORENTINI, 1995, p. 5).

Embora o trabalho com RP em todos os anos desenvolva habilidades necessárias para o bom desempenho no ensino de matemática e sejam um requisito para provas de larga escala, sabemos que grande parte dos nossos estudantes não são capazes de realizar RP com operações fundamentais simples, cálculos aritméticos, números naturais, seguindo instruções claras e bem-definidas (OECD, 2013). De acordo com a Organização para a Cooperação e Desenvolvimento Econômico (OCDE), que organiza o Pisa a cada três anos, apenas 2% dos estudantes brasileiros demonstram ser capazes de resolver problemas complexos em avaliações externas, e 68,1% se encontra no nível 1 ou abaixo dele (OECD, 2013).

Quando comparado com a média dos países membros da OCDE, como demonstra a Figura 2, o desempenho do Brasil está muito aquém do esperado. A média da OCDE permanece próxima a 500, e a do Brasil, entre 377 e 389 nos últimos nove anos (entre 2009 e 2018). Houve evolução nos índices, mas isso não retira o país das piores posições no ranking relativo à matemática do exame.

Figura 1 - Média de matemática - Brasil x OCDE - avaliação do PISA 2019



Fonte: adaptada de Moreno e Oliveira (2019).

2.4 PC, definição e aplicação

O conceito de PC é abrangente e não apresenta uma definição única, uma vez que cada estudioso tenta conceituá-lo a partir de sua área de atuação. Além disso, deve-se levar em conta que a resolução dos problemas a serem resolvidos pode ser questionável e analisável.

Portanto, é necessário delimitar o escopo e a complexidade do PC, conceituando e definindo uma estrutura clara que servirá de apoio no desenvolvimento pedagógico e na avaliação do PC, especialmente no âmbito dos Ensinos Fundamental e Médio em matemática.

A Sociedade Brasileira de Computação, instituição de referência na área para estudantes, profissionais e pesquisadores de computação do Brasil, cita que o PC:

O Pensamento Computacional se refere à capacidade de sistematizar, representar, analisar e resolver problemas. Apesar de ser um termo recente, vem sendo considerado como um dos pilares fundamentais do intelecto humano, junto com

leitura, escrita e aritmética, pois como estes, serve para descrever, explicar e modelar o universo e seus processos complexos. (RAABE, 2017, s.p.).

Papert (1980), em seu livro *Mindstorms: Children, Computers and Powerful Ideas*, traz as primeiras discussões acerca dos benefícios de ensinar habilidades computacionais e propõe incluir a computação no Ensino Infantil, pois tal ação se torna mais significativa quando as próprias crianças produzem seus projetos, afinal eles passam a ter maior valor, proximidade e aceitação, se houver reflexão de maneira cuidadosa em cada etapa do processo (PAPERT, 1980).

Papert (1980) expõe ainda que ajudou na elaboração de uma linguagem de programação chamada Logo, cujo propósito é auxiliar as crianças e os jovens a estruturar a maneira pela qual pensariam, articulariam e resolveriam problemas, inserindo-se como parte de seu desenvolvimento. Essa linguagem foi projetada com fundamentos de fácil compreensão e pode ser utilizada por uma criança que não tenha muitos conhecimentos tecnológicos. Apesar disso, não deixa de ser rica e ampla o suficiente para níveis avançados, de modo que uma pessoa com mais conhecimentos possa usufruir de seus recursos, o que fez com que a linguagem fosse muito usada como ferramenta no ensino regular. Com isso posto, o autor a firma:

No ambiente Logo, a criança, mesmo em idade pré-escolar, está no controle – a criança programa o computador. E, ao ensinar o computador a “pensar”, a criança embarca em uma exploração sobre a maneira como ela própria pensa. O foco dos estudos de Piaget foi o “sujeito epistêmico”, ou seja, o estudo dos processos de pensamento presentes no indivíduo desde a infância até a idade adulta. Pensar sobre modos de pensar faz a criança tornar-se um epistemólogo, uma experiência que poucos adultos tiveram. (PAPERT, 1986, p. 25).

Tal pensamento construtivista do autor baseia-se nas teorias de Piaget (1954), segundo o qual a criança constrói, de maneira ativa, seu próprio conhecimento quando o realiza por meio de experiência, *aprendendo e fazendo*. Ainda que a teoria de Piaget seja desenvolvida para explicar como o conhecimento é formado na mente do indivíduo, Papert (1980) amplia os conceitos enquanto os aplica à computação e suas linguagens, de modo a se concentrar nas formas pelas quais as construções internas são suportadas por construções do mundo, inclusive por meio do uso de computadores e robótica. Nesse momento as crianças podem explorar seus interesses por meio da tecnologia, e o aprendizado se dá pelo exercício das habilidades cognitivas e pela resolução de problemas propostos.

Assim, é preciso que o aluno seja ativo em seu próprio desenvolvimento: deve construir, realizar uma *aprendizagem mão na massa*, para aprender fazendo, pois desse modo o ensino será efetivo, e a criança ou o jovem estará sempre motivado e de posse do seu objeto de aprendizagem. (PAPERT, 1980).

Nesse cenário, ainda que a expressão “pensamento computacional” já existisse, foi com Jeannette Wing (2006) que o termo realmente ganhou visibilidade e notoriedade.

A combinação do pensamento crítico com os fundamentos da Computação define uma metodologia para resolver problemas, denominada Pensamento Computacional. É uma distinta forma de pensamentos com conceitos básicos da Ciência da Computação para resolver problemas, desenvolver sistemas e para entender o comportamento humano, habilidade fundamental para todos. (WING, 2006).

Jeannette Wing foi professora do curso de Ciência da Computação da Columbia University nos Estados Unidos. De acordo com ela, o PC envolve a resolução de problemas, a concepção de sistemas e a compreensão do comportamento humano, com base nos conceitos fundamentais da ciência da computação. Além disso, ela propõe que o PC aborde problemas que possam ser resolvidos por meio do projeto de um sistema em seu entorno, para que as soluções sejam representadas de tal forma que seja possível o efetivo processamento das informações pertinentes (WING, 2010, p. 3). Ou seja, essas ferramentas propõem que, dado um problema dotado de certo grau de dificuldade, conseguiríamos resolvê-lo facilmente dentro desses parâmetros. Wing, no decorrer dos seus estudos, verificou que o conceito de PC é muito mais amplo do que ela mesma sugeria, o que lhe permite aprimorar suas definições: “processos de pensamento envolvidos na formulação de problemas e as suas soluções de modo que as mesmas são representadas de uma forma que pode ser eficazmente executada por um agente de processamento de informações” (WING, 2010, p. 81).

Em outro estudo, Wing (2016) estabelece que qualquer coisa que possa ser resolvida valendo-se do processo de abstração (em que se decide quais detalhes devem ser ressaltados e quais devem ser ignorados) pode ser apontado como PC, ressaltando que essas habilidades não deveriam ficar restritas a estudantes de computação, mas ampliadas de forma geral. Nesse sentido, sustenta posteriormente que o PC é parte do processo de abstração, em que se faz necessária a escolha de uma habilidade para resolução de problemas (WING, 2016). Tais processos são inerentes ao cotidiano, e nós nem percebemos quando os aplicamos a situações corriqueiras, como escolher a melhor maçã em meio a uma cesta cheia, ou qual a melhor fila do banco em que aguardar ser atendido, ou ainda qual o melhor trajeto para uma viagem.

São os processos de pensamento envolvidos na formulação de um problema e que expressam sua solução ou soluções eficazmente, de tal forma que uma máquina ou uma pessoa possam realizar. É a automação da abstração e o ato de pensar como um Cientista da Computação. (WING, 2014, p. 26).

Assim, os processos de pensamento envolvidos na RP e suas soluções só ocorrem de forma efetiva se realizados por um agente de processamento de informações (WING, 2010), o que significa que o PC traz como sua essência que o discente deve pensar como cientista da computação ao ser confrontado com um problema. Mas como pensa um cientista da computação? Quando observa um problema, esse profissional procura encontrar uma solução para resolvê-lo, havendo uma infinidade de olhares possíveis a serem aplicados para que a resolução aconteça.

Pela perspectiva da matemática, para responder a um dado questionamento, devemos propor estratégias claras e objetivas. Também na computação, é necessário descrever claramente o processo, propiciando a modelagem da resolução, extrapolando os procedimentos para uma infinidade de problemas de mesma categorização, a partir de uma única técnica ou algoritmo. Isso é válido tanto para as questões mais simples até para as mais complexas.

O conceito de RP é parte integrante do PC, em que um problema pode ser dividido em etapas menores, de forma que possa ser compreendido melhor pelos estudantes. Essa divisão também constitui uma maneira de estruturar o problema logicamente, tornando-se assim possível utilizar recursos computacionais para sua resolução. Todavia, é importante ressaltar que o objetivo é melhorar o entendimento e o raciocínio lógico, e não apenas “terceirizar” a resolução para um dispositivo eletrônico.

Não se trata, de forma alguma, de fazer com que as pessoas executem comandos como computadores, mas sim de utilizar os recursos computacionais com criatividade e estratégia com o intuito de identificar, sistematizar e analisar a resolução de problemas com passos claros e de maneira eficiente, correlacionando ambas as técnicas. Esses argumentos deram uma nova perspectiva às relações que se estabelecem entre computadores e seres humanos, dando origem a uma onda de pesquisas sobre PC.

De acordo com Blikstein (2008, p. 12), “pensamento computacional é saber usar o computador como um instrumento de aumento do poder cognitivo e operacional humano, em outras palavras, usar computadores, e redes de computadores, para aumentar nossa produtividade, inventividade e criatividade”. A maneira pela qual a tecnologia é utilizada nas escolas é classificada pelo autor como “adestramento digital”, em que o aluno apenas se vale dos recursos tecnológicos para encontrar e utilizar mais rapidamente as informações, ficando muito aquém do potencial de uso efetivo desse instrumental. Complementa o autor que é possível utilizar as conjecturas do pensamento computacional em atividades diversas, ainda que sem o uso do computador, com as chamadas atividades “mão na massa”, que desenvolvem habilidades e competências do aluno para *se pensar computacionalmente*, organizando os tipos de tarefas e suas resoluções de maneira mais rápida e efetiva (BLIKSTEIN, 2008).

Outro autor que exprime, em suas produções, pensamentos necessários e alinhados aos objetos de que tratamos nesta pesquisa é DiSessa (2018), o qual constata que diferentes meios de comunicação possibilitam novos pensamentos e organizações intelectuais sobre diferentes áreas, promovendo a aprendizagem. Ele reflete ainda sobre como ensinar o PC em uma perspectiva na qual as pessoas não sejam apenas consumidoras de artefatos computacionais, mas sim criadoras daquilo que o autor define como Alfabetização Computacional, compreendida em três eixos relevantes: social, cognitivo e material. Nesse processo de letramento e instrumentalização da linguagem computacional e da articulação de seu pensamento, instrumentaliza-se a ciência da computação para resolver problemas (DISESSA, 2021).

Campana (2017) utiliza a construção de jogos de tabuleiro para o ensino de matemática, os quais são produzidos por alunos e professor especialista no formato físico e depois transpostos para o formato digital. Esse processo de construção e de utilização de jogos relaciona-se com a ideia de DiSessa (2018), de acordo com quem a Alfabetização Computacional consiste em criar, entender, seguir regras de depuração para o desenvolvimento de habilidades do PC, em que se desenvolve um conjunto de regras que podemos chamar de *instruções* ou *desenvolvimento de algoritmos*.

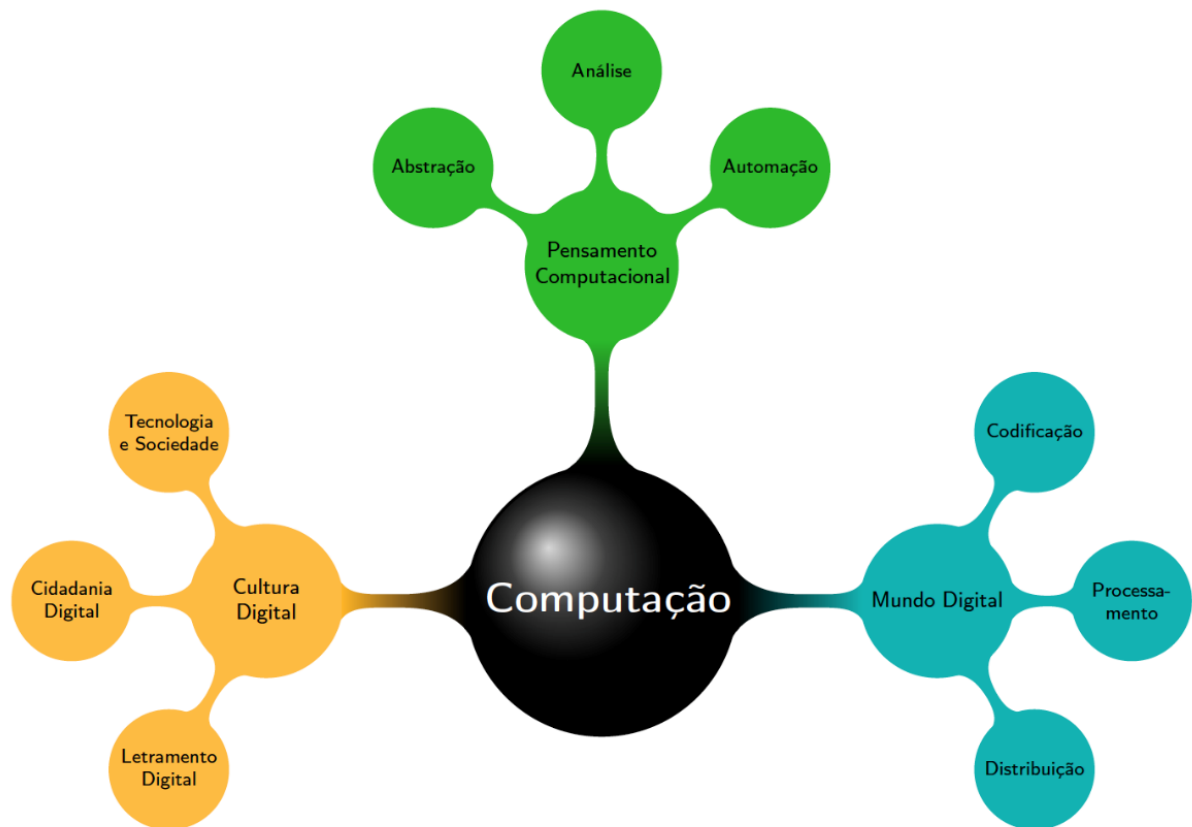
Quando os alunos fazem uso desses jogos de tabuleiro, relacionados ao estudo de Campana (2017), ou a qualquer outro jogo, seguem instruções, um passo a passo, para atingir o resultado desejado, qual seja, resolver o problema do jogo, como:

- Lance o dado.
- Ande para o lado positivo, se no seu dado a operação foi maior que zero.
- Caso contrário, ande para o lado negativo.

Fazemos referência a isso como um conjunto de instruções de algoritmo que será base desse trabalho como uma Alfabetização Computacional.

Cabe aqui mencionar que a Sociedade Brasileira de Computação (SBC) tem realizado uma série de estudos para indicar e recomendar a inserção do PC na Educação Básica, pois fica cada vez mais clara e notória a necessidade e a importância do desenvolvimento do estudante nesse assunto. Sendo assim, a SBC propõe uma definição para PC, com diretrizes e parâmetros específicos ao ensino de Computação na Educação Básica, sendo a abordagem didática dividida em três eixos, os quais, por sua vez, compartmentam-se em outros três tópicos cada: Cultura Digital (tecnologia e sociedade, cidadania digital e letramento digital), Pensamento Computacional (abstração, análise e automação) e Mundo Digital (codificação, processamento e distribuição), como mostra a Figura 2.

Figura 2 - Eixos da abordagem didática da computação na Educação Básica



Fonte: Segmentos da computação – Proposta da SBC (MINISTÉRIO DA EDUCAÇÃO, 2017, p. 11).

À esquerda da imagem, encontramos o eixo da Cultura Digital, que aborda diferentes linguagens e conceitos, entre eles o letramento digital. Nessa linha não importa somente saber manusear a tecnologia, mas sim utilizá-la de forma crítica e ética dentro do mundo digital, sistematizando a informação de diferentes formas e com diversas finalidades. A Cultura Digital vem estabelecer uma ligação, de forma interdisciplinar, da tecnologia com outras áreas do conhecimento, pautada sempre em questões éticas e reflexivas. Nesse eixo, ensinam-se os cenários e os objetos dos ambientes virtuais que podem gerar embates e questionamentos éticos, como *fake news*, uso indevido de imagens, violação de direitos autorais e artísticos pela cópia e pirataria de arquivos, *cyberbullying*, entre outros temas que devem ser postos em perspectiva e cujas discussões são desenvolvidas coletivamente no ambiente escolar, para a construção da ética, da cidadania e da criticidade.

À direita há o eixo do Mundo Digital, que se ocupa de compreender como a sociedade utiliza dispositivos tecnológicos, de modo que não apenas se valha dos recursos computacionais disponíveis, mas entenda sua dinâmica para conseguir se apropriar de maneira inteligente e crítica do seu potencial. Os conceitos de codificação, processamento e distribuição da informação estão interconectados de forma inédita na história humana, pois não houve antes tanta informação

disponível, o que só é possível dada a capacidade de processamento dos computadores e a codificação de *softwares*.

Para finalizar, temos o eixo Central, em que se encontra o PC — dividido em abstração, análise e automação. Na pesquisa da SBC, o PC se caracteriza como uma das bases da proposição de inserção desse conteúdo na Educação Básica. Desse modo, o PC, o algoritmo e a RP estão interligados, uma vez que o algoritmo acaba se constituindo como uma etapa descritiva de todo o processo para resolver problemas, que se dá de maneira organizada e voltada ao desenvolvimento de habilidades e capacidades. Além disso, hoje a resolução de problemas complexos passa mandatoriamente pelos algoritmos executados em computadores, e, por meio dos modelos computacionais, podemos parametrizar situações e enxergar uma resolução.

2.4.1 Sintetizando conceitos e definições do PC

Após toda a reflexão engendrada, dispomos do quadro 1, para apresentar uma síntese sobre o PC bem como as definições dos principais autores abordados nesta pesquisa, com o objetivo de ilustrar os benefícios que o conhecimento sobre o PC pode proporcionar na educação.

Quadro 1 - Definição de autores acerca de PC

Autor	Definição de PC
Seymour Papert (1980)	Entende-se que, por meio do construtivismo e da linguagem de programação, surgiu o PC, como uma ferramenta que auxilia a construção do conhecimento. Os computadores trabalham para a resolução de problemas aplicados em diferentes áreas.
Paulo Blikstein (2008)	O PC é um instrumento que potencializa as habilidades de cognição dos seres humanos, ou seja, ao utilizar o computador e suas funcionalidades, incrementa a produção, a criatividade e a iniciativa do ser humano.
Jeannette Wing (2006, 2010, 2014)	A autora destaca que o PC é uma junção do pensamento crítico e dos fundamentos da computação em uma metodologia focada na RP. Outro ponto importante é que a RP não se restringe apenas aos pilares da ciência da computação, e sim é vista como uma habilidade que deve ser desenvolvida por todos (2006). A utilização eficaz do PC, por meio de ambientes de programação, foca a resolução de problemas a partir de suas formulações (2010). O conceito de abstração está atrelado fortemente a soluções de problemas, desta forma o PC pode ser entendido como uma forma de conseguir extrair o entendimento da abstração, seja

	por meio de um cálculo manual, seja por um <i>software</i> específico (2014).
Linda Liukas (2015)	O objetivo do desenvolvimento de um <i>software</i> é automatizar uma tarefa repetitiva. Mas, antes disso, é importante destacar que <i>softwares</i> não resolvem tudo, pois são desenvolvidos por pessoas, e elas precisam ter bem fundamentados conceitos como lógica, reconhecimento de padrões, pensamento algorítmico, decomposição e abstração.
Christian Brackmann (2017)	O PC pode ser compreendido como uma forma de RP nos mais variados campos do conhecimento humano. Sendo assim, o problema é identificado, e sua posterior resolução pode ser realizada em atividades individuais ou coletivas. Porém, é importante que a forma de resolução seja compreendida o mais fácil possível, para que possa ser executada de maneira eficaz tanto por uma pessoa quanto por um programa de computador.
Andrea DiSessa (2018)	Constata-se que diferentes meios de comunicação possibilitam novos pensamentos e organizações intelectuais sobre diferentes áreas, promovendo a aprendizagem e alfabetização computacional.
Sociedade Brasileira de Computação	Para a SBC, o PC menciona a capacidade de sistematizar, representar, analisar e resolver problemas. Ele é um pilar necessário para o desenvolvimento do intelecto em áreas como leitura, escrita ou aritmética, servindo como base para explicar e modelar o universo e seus processos complexos.

Fonte: desenvolvida pela autora (2020).

A ideia central é ensinar alunos da Educação Básica ou qualquer pessoa que seja a aplicar conceitos e elementos semelhantes para a RP, a fim de, dessa maneira, enxergar novas perguntas que possam ser observadas dentro de toda área do conhecimento. Assim, para o desenvolvimento deste trabalho, empregaremos duas das principais características do PC observadas e demonstradas na literatura: a decomposição e o algoritmo. Os estágios (Figura 3) para o PC são caracterizados pelos autores como:

Figura 3 - Estágios para o PC

Fonte: desenvolvida pela autora (2020).

Entre todas as definições anteriores, a que mais estabelece diálogo com esta pesquisa consiste em uma junção de conceitos essenciais presentes em mais de um autor, ou seja, a de que o PC é um suporte fundamental para RP de maneira eficiente, por meio do uso de computadores ou não, reutilizando, sempre que possível, as resoluções já desenvolvidas em diferentes situações, de modo que o indivíduo/aluno possa pensar e agir aplicando suas habilidades lógicas.

2.5 Habilidades do PC

Com base nas reflexões e nas discussões sobre o tema, surge um novo olhar para aprofundar pesquisas nesse assunto de PC e suas implicações cognitivas educacionais. A seguir, é apresentada uma breve explanação de cada uma das características do PC, já que elas servirão de base para o desenvolvimento do presente trabalho.

2.5.1 Decomposição

O processo de decomposição, que constitui um dos pilares do PC, pode ser considerado um conhecimento já explicitado anteriormente na matemática, na medida em que torna possível que cálculos sofisticados e complexos sejam realizados — não somente aqueles que, de são

avancados, apenas recentemente tiveram sua resolução tornada mais ágil nas últimas décadas, contando com o apoio de *softwares* e *hardwares* para tanto.

Sob essa perspectiva, a maior parte da literatura trata a decomposição como uma etapa comum na RP, determina que a decomposição é uma das habilidades do PC sem trazer detalhes ou especificidades a respeito de como ela acontece, tampouco se fala sobre a melhor forma de utilizá-la na RP, ou ainda como pode ser mais ou menos eficaz em seu uso.

Consideramos, então, que a decomposição seja a divisão de um problema em partes menores, tornando-se mais fáceis de serem resolvidas. A princípio, tentamos vê-lo por uma perspectiva holística, depois o dividimos em partes menores. Para ser entendido, o problema é dividido em três etapas, sendo o reconhecimento da associação dos itens, selecionando um método para reduzir parcialmente situações indesejadas e investigar a eficácia do método em outros problemas (RICH, 2019). Assim quando realizadas subdivisões as entradas, saídas e outras relações entre o problema central e subproblemas tornam-se conhecidas e gerenciáveis. Desta forma, a decomposição é uma das habilidades primordiais e iniciais do PC; fazer o seu uso correto pode melhorar todo o processo do PC e a RP.

Um aspecto importante e largamente citado na literatura sobre PC é seu potencial de irradiar conhecimentos que não apenas se limitem a resolver problemas no plano da ciência da computação, mas também se apliquem, por meio de seus pilares bem-fundamentados, a outras áreas, inclusive a situações de cotidiano, de modo que possam ser de grande valia para a Educação Básica, por exemplo (DA SILVA; MENEGETTI, 2019).

Nota-se, porém, que não se trata de algo intuitivo somente: o pilar do PC é o processo cognitivo, de modo que ele não se resume apenas ao desenvolvimento de produtos tecnológicos, como geralmente é compreendida a computação de maneira mais genérica na sociedade (SELBY; WOOLLARD, 2013). É importante destacar que, embora o produto tecnológico desenvolvido seja o que de mais aparente reflita o emprego de conhecimentos científicos, as bases que o sustentam são justamente os conceitos do PC, como é o caso da decomposição.

Um aspecto importante da decomposição é o reuso de uma solução para resolver um mesmo problema, contribuindo para agilizar e facilitar o desenvolvimento de respostas ágeis que garantam solidez na resposta final, o que é muito importante (OLIVEIRA, 2019).

Aspectos do nosso cotidiano, como realizar uma multiplicação ou divisão, têm, em seu âmago, o conceito da decomposição embutido, pois cálculos matemáticos complexos são muitas vezes involuntariamente divididos em partes menores para que sua resolução seja simples e que o resultado seja atingido e interpretado de maneira correta. Selby e Woollard (2013) definem que a decomposição é necessária para que se possa lidar com tarefas e sistemas complexos. Um

aspecto implícito desses primeiros parágrafos é a associação muito próxima entre o PC e a matemática, tanto pelo processo de decomposição como por outros constituintes do PC, como a algoritmização e a abstração (DA SILVA; MENEGETTI, 2019).

Esses pilares do PC podem e devem ser usados de maneira a abranger aspectos do conhecimento que possam colaborar, de forma ampla, para a Educação Básica. A decomposição aqui citada pode ser usada em diversos aspectos do aprendizado, e não somente na matemática e na computação. Nos tópicos a seguir, o objetivo é este: delinear possibilidades do uso da decomposição de maneira mais abrangente.

2.5.2 Decomposição na matemática

De modo mais abrangente, temos que na matemática a decomposição está associada também à composição, ou seja, é necessário decompor e recompor os resultados. Em cálculos matemáticos, esse procedimento é necessário para que o aluno compreenda todo o processo de repartição e junção dos resultados. Assim, o aprendiz é levado a enxergar o problema em partes menores (decomposição) e, por sua vez, também o todo (composição).

Na Base Nacional Comum Curricular (BNCC), fica explícita tal preocupação com a decomposição, vemos diferentes habilidades englobando o assunto, como nos exemplos citados abaixo:

- Reconhecer e identificar a decomposição de números naturais nas suas diversas ordens.
- Reconhecer a composição e a decomposição de números naturais em sua forma polinomial (BRASIL, 2016).

2.5.3 Decomposição no PC

Na perspectiva do PC e do seu uso na resolução de problemas computacionais, a decomposição pode ser vista por diferentes matizes, como ao criar uma função que realize uma determinada operação. Por exemplo, é muito comum, em sistemas digitais, a verificação de validade de um CPF; nesse contexto, o desenvolvedor cria uma função que possui somente essa designação, isto é, verificar se um número digitado é válido ou não. Note que essa resolução pode ser depois aplicada em outros *softwares* que tenham objetivos e contextos diferentes do exibido. Assim, pode-se verificar se o número de CPF é válido para uma transação comercial, financeira ou somente cadastral.

Quando buscamos na literatura trabalhos relacionados à habilidade da decomposição no PC, percebe-se que existe uma gama de autores, como Kong (2019), que, embora tratem desse conceito, não o fazem dando-lhe enfoque central, pois têm como foco o próprio PC. Considerando essa abordagem, este trabalho busca abordar como o aluno observa e desenvolve o pensamento lógico e a decomposição ao resolver problemas.

Entre os trabalhos relacionados à decomposição, podemos citar Rich *et al.* (2018). Os autores apresentam as fases de construção da aprendizagem por meio da prática da decomposição.

Rich, Egan e Ellsworth (2019) apresentam uma estrutura para que os professores estejam mais preparados para auxiliar seus alunos a desenvolverem problemas complexos utilizando a decomposição, de modo que essa capacidade possa ser medida. Os autores acreditam ter descrito o mistério da caixa-preta da decomposição, sendo vista como um processo de três etapas, que são: “(1) categorizar elementos potenciais, (a) identificar elementos substantivos, (b) identificar relações entre os elementos, (2) empregando diferentes estratégias para executar a decomposição, (3) avaliação interativa da utilidade de uma decomposição específica” (RICH; EGAN; ELLSWORTH, 2019, p. 4).

Kwon e Cheon (2019) estudaram os processos dos alunos em programação com Scratch e analisam dois aspectos principais: a decomposição do problema e o desenvolvimento do programa. Introduziram os conceitos de *loops*, dados, condições e padrões, e juntamente examinaram o desenvolvimento dos programas. Concluíram que os alunos tinham dificuldade com as condições e os operadores nos programas, e que eles precisavam testar seus programas com muito mais frequência para ter chance de solucionar o problema, mesmo aplicando a decomposição adequadamente.

A quebra de problemas com funcionalidade é caracterizada por Wing (2006) como parte do PC. A realização da decomposição se faz necessária toda vez que nos deparamos com problemas grandes, sistemas complexos ou tarefas complexas. Pode-se dizer que a decomposição não é apenas um componente sugerido do PC, mas também uma técnica utilizada na educação e em diferentes áreas do conhecimento para RP.

Sabendo, então, que a decomposição é uma habilidade extremamente importante para se resolver problemas matemáticos ou de outra natureza, e considerando ainda que temos poucos estudos sobre essa habilidade e sobre como ela se desenvolve de forma gradativa, detalharemos experiências cotidianas dos alunos neste trabalho.

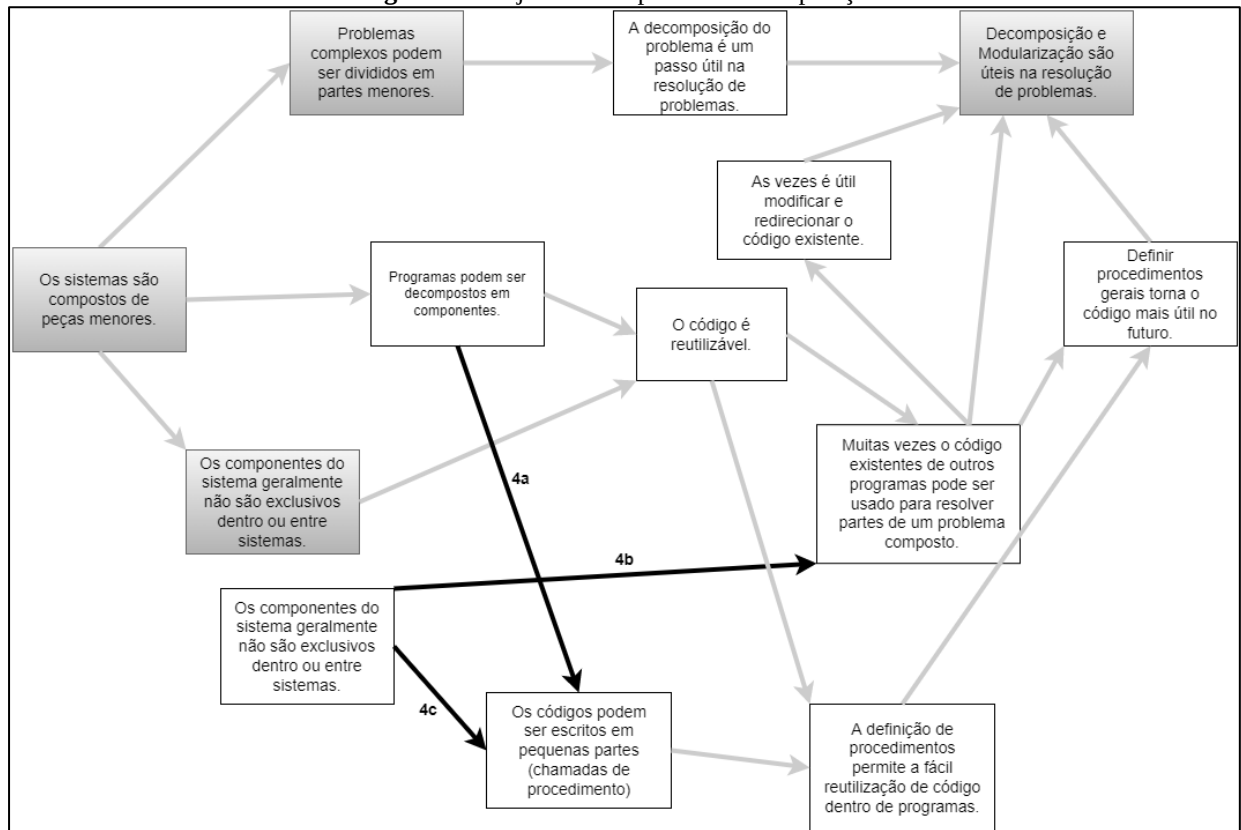
Rich (2019) realiza um estudo com diferentes abordagens, identificando e classificando a decomposição nas seguintes categorias:

- Decomposição estrutural, que consiste em dividir o problema em subproblemas, isto é, em partes menores e gerenciáveis, resolvendo separadamente cada um.
- Decomposição funcional, em que cada parte pode estar relacionada com outra parte, sempre considerando as funções que ocupam e a relação entre elas. Assim, a decomposição acontece observando sua função de execução, sendo analisada sempre separadamente. Ou seja, os objetos são separados em subgrupos, unidos com funções apropriadas.
- Decomposição sequencial, em que se caracteriza a ordem de todas as operações dentro do processo. Nessa categoria, as etapas são importantes.
- Decomposição por dependência, que, ao dividir o problema em subproblemas, leva em consideração a relação e a dependência entre as outras partes.

Cada tipo de decomposição é mais adequado a problemas específicos. Às vezes a decomposição acontece antes da abstração; às vezes, depois. Insere-se no processo de RP, então é necessário conhecer suas várias nuances e tipologias para que possa ser mais bem utilizada.

Embora várias abordagens para o ensino de computação tenham sido estudadas, existem poucas pesquisas que foquem a decomposição como objeto de aprendizagem. Em recente estudo sobre decomposição, Rich (2018) aborda o tema em três etapas. Em um primeiro momento, realiza uma revisão de literatura de mais de cem publicações em educação em ciência da computação, focando as trajetórias de aprendizagem desenvolvidas e organizando-as por conceito ou prática, incluindo sequência, repetição, condicionais, decomposição e depuração. Em seguida, sistematiza os conteúdos desses objetivos de aprendizagem e finaliza, por meio das informações encontradas na literatura, criando estruturas conceituais e produtivas que os estudantes poderiam utilizar, por consenso, para melhorar o seu processo de decomposição (RICH, 2018).

Nesse estudo Rich (2018) realiza estruturas de conexões apoiadas pela literatura, construtivismo e *framework*, finalizando com a junção de diferentes composições e formando a trajetória completa da decomposição. A autora ainda propõe uma abordagem coletiva, que faz com que o estudante possa se identificar com o processo de decomposição por meio de experiências vividas anteriormente, não apenas no ambiente computacional, afinal fora dele isso também é possível. Assim, pode-se impulsionar a compreensão e a prática da decomposição dentro do currículo de matemática ou em qualquer campo de estudo.

Figura 4 - Trajetória completa da decomposição

Fonte: adaptada de Rich *et al.* (2018).

A Figura 4 está estruturada dentro da trajetória da decomposição, e nela é claro observar as duas dimensões propostas: a decomposição de problemas, que acontece no processo de cima para baixo; e a criação de códigos, que faz o processo inverso. Apesar disso, elas se cruzam, não estando totalmente separadas em seu percurso, pois dependem uma da outra.

Dessa forma, quando o estudante pensa em como realizar a decomposição de um problema ou de um programa em partes, essa ação permite que ele reflita sobre como pode escrever partes de códigos. As ações estão interligadas, e foi com base nelas que as atividades desta tese foram pensadas, a princípio separadas, porém, em sua construção e realização, elas se encontram em determinadas partes do processo de resolução.

Observemos o Problema 1, descrito no Quadro 2, de nível fácil, voltado para os anos finais do ensino fundamental, que pode ser resolvido com lógica matemática. Podemos utilizar diferentes habilidades do PC para resolvê-lo, porém, neste primeiro momento, focaremos a habilidade da decomposição. Lembramos que a habilidade da decomposição na RP pode ser aplicada em problemas matemático complexos ou simples, como o descrito aqui.

Quadro 2 - Problema envolvendo o conceito de decomposição

Problema 1
Três homens querem atravessar um rio. O barco suporta no máximo 130 kg. Eles pesam 60 kg, 65 kg e 80 kg. Como devem proceder para atravessar o rio sem afundar o barco? Considere que o barco precisa de um condutor, pois não atravessa o rio sozinho.

Fonte: adaptada de Niederauer e Aguiar (2007).

A decomposição de problemas geralmente nos leva ao reconhecimento e à generalização de padrões, e dessa forma à capacidade de representar bons algoritmos. Interligando ações e decompondo a resolução, isto é, dividindo-a em partes, o problema será resolvido de forma organizada e eficiente. Por exemplo, é possível separar os dados. Sob esse viés, vejamos como poderia ser aplicada a decomposição:

Quadro 3 - Decomposição do problema

Qual o problema a ser resolvido?	Dividindo em grupos menores de problemas
Que três homens atravessem o rio.	1 - Homem 1 e Homem 2 vão para o lado B do rio. 2 - Homem 1 volta para o lado A do rio (fica do lado A). 3 - Homem 3 vai para o lado B do rio. 4 - Homem 2 volta para o lado A do rio. 5 - Homem 2 e Homem 1 vão para o lado B do rio.

Fonte: desenvolvida pela autora (2020).

Ao realizar agrupamentos de modo que a soma do peso das pessoas presentes dentro do barco não ultrapasse 130 kg, já que essa é uma restrição proposta pelo problema, é possível criar etapas para se resolver o problema, levando em consideração também a segunda condição de que o barco não pode navegar sozinho. Nesse caso, as cinco etapas descritas, no quadro 3, resolvem essa situação, ou seja, criam, durante a resolução, uma escrita de códigos. Ao analisar como os homens poderiam chegar ao outro lado do rio, as idas e vindas constituem os “grupos ou subgrupos de problemas”, realizando subseções com ordem para executar as tarefas, ou seja, estruturam uma decomposição sequencial, em que a ordem das tarefas executadas é levada em consideração, facilitando assim a resolução e tendo o resultado mais efetivo.

Quadro 4 - Construção do hexágono

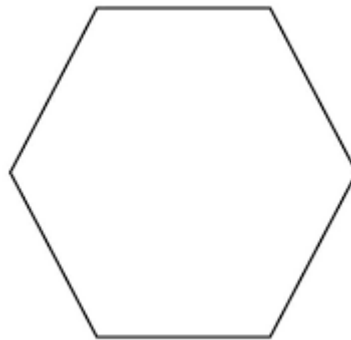
Problema 2

Desenhe um hexágono utilizando a decomposição.

Fonte: desenvolvida pela autora (2020).

A matemática subdivide-se dentro do seu grupo de estudo; podemos citar a álgebra, a aritmética e geometria. No segundo problema, quadro 4, a atividade estará centrada no campo da geometria, que realiza o estudo do espaço e forma. A atividade é simples: desenhar um hexágono (Figura 5) de maneira espontânea. Nesse momento é necessário abstrair, em nossas mentes, a imagem de um hexágono para realizar a construção.

Figura 5 - Hexágono



Fonte: desenvolvida pela autora (2020).

Quadro 5 - Decomposição do hexágono

O que o problema quer?	Divisão em grupos menores de problemas
O desenho de um hexágono.	1 - Desenhe uma linha reta horizontal. 2 - Desenhe uma linha diagonal à sua direita. 3 - Desenhe uma linha diagonal à sua esquerda. 4 - Desenhe uma linha reta na horizontal. 5 - Desenhe uma linha diagonal à sua esquerda. 6 - Desenhe uma linha diagonal à sua direita.

Fonte: desenvolvida pela autora (2020).

Ao realizar mentalmente uma abstração do hexágono, conseguimos decompor, quadro 5, o problema em pequenos grupos para poder solucioná-lo parte a parte. Não podemos confundir

decomposição, que consiste em dividir o problema em tarefas ou partes menores, com algoritmo, que descreve a solução detalhada de um problema a ser resolvido. Nesse caso, podemos dizer que o problema em questão é a construção de um hexágono, que nada mais é do que a junção de segmentos de retas e giros. Existe, então, uma repetição de comandos alternados, podendo ser chamada de “padrão”⁴, pois o movimento se repete diversas vezes. Essa sequência de passos está sempre interligada a comandos anteriores, determinando um tipo de relação, como dependência.

Quadro 6 - Soma numérica

Problema 3
Encontre a soma de todos os números entre 1 e 300. Como podemos dividir esse problema em equações menores?

Fonte: desenvolvida pela autora (2020).

Uma maneira de dividir o problema em partes menores seria quebrando a equação, quadro 7, em equações menores, por exemplo, somando os números na sequência. Vejamos:

Quadro 7 - Soma de números

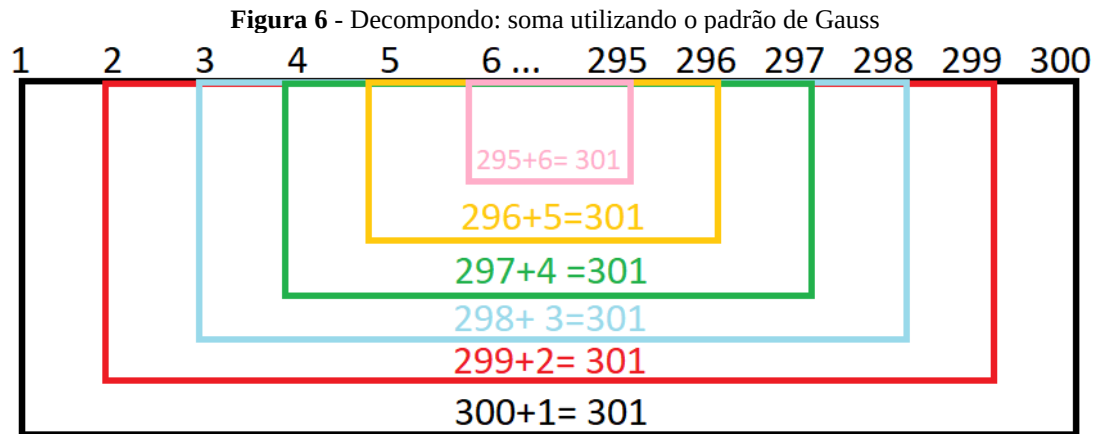
$1+2 = 3$	$3+4 = 7$	$5+6 = 11$
-----------	-----------	------------

Fonte: desenvolvida pela autora (2020).

Assim, não seria necessário somar todos os números de uma única vez; sabemos, aliás, que essa ainda não seria a melhor maneira, pois o tempo despendido seria muito grande e seria possível ocorrer perda do raciocínio durante a execução da atividade.

Podemos ainda tentar outra solução, do problema 3, quadro 6, com conjuntos diferentes de combinações numéricas, dividindo os números em parcelas menores. Nesse caso serão somados o último e o primeiro número da sequência, o segundo e o antepenúltimo, e assim sucessivamente, propondo assim uma decomposição diferente para um mesmo problema, conforme mostra a Figura 6.

⁴ Esse conceito será descrito com mais detalhes no item 2.5.4.



Fonte: desenvolvida pela autora (2021).

Essa maneira de resolver problemas é conhecida como soma de Gauss, em que um número está relacionado a outro em posição oposta a ele. No caso em questão, a soma de ambos sempre será 301. Tal método foi desenvolvido por Carl Friedrich Gauss, matemático alemão que conseguiu resolver rapidamente a soma dos números naturais entre 1 e 100, chegando à definição de que a soma de dois números equidistantes dos extremos de uma progressão aritmética finita é igual à soma de seus extremos.

Decompor um problema em partes envolve uma série de etapas e conhecimentos, que contribuem para o ato em si de decompor o problema. A decomposição pode ser repetida e reutilizada ao ser empregada em problemas simples ou complexos por meios de padrões e algoritmos. Podemos aprender muito ao analisar diferentes formas de decompor um mesmo problema e verificar quais técnicas podem facilitar tais resoluções. Assim, sabemos que a decomposição pode e deve ser utilizada como estratégia e mecanismo para facilitar resoluções.

2.5.4 Reconhecer padrões

Após decompor problemas de ordem complexa, pode-se realizar uma busca por características e semelhanças desses subproblemas, as quais chamaremos de “padrões”⁵, em cada segmento do problema proposto. Reconhecer padrões é uma etapa importante, pois nos remete a identificar tudo que é comum dentro de diferentes situações; padronizar nos permite resolver problemas complexos de forma mais rápida e eficiente, pois, quando se conhece o padrão, é possível utilizá-lo para replicar a resolução em diferentes problemas.

⁵ Identificar semelhanças ao se resolver problemas e utilizar esses padrões encontrados em diferentes resoluções.

Assim, vamos em busca de itens que compartilhem semelhanças em cada ordem na resolução do problema. Nem sempre haverá elementos semelhantes, porém, ainda assim, essa deve ser uma etapa do processo. Ao realizar o reconhecimento de padrões, podemos encontrar semelhanças como cor, forma, largura e espessura, além de esgotar os recursos necessários que esgotem todas as possibilidades para que ocorra uma padronização do problema. Quanto maior a quantidade de padrões encontrada, mais fácil e rápida será a resolução do problema, afinal, uma vez reconhecido o padrão, podemos trabalhar em soluções comuns entre os problemas.

Essa capacidade de reconhecer partes utilizadas em situações passadas, e que serão utilizadas em situações futuras, é reforçada por Kolodner (2011). O autor afirma que as partes funcionais podem ser utilizadas para resolver problemas simples ou combinadas de diferentes maneiras para se resolver novos.

No Problema 3, citado anteriormente, podemos reconhecer alguns padrões ou tendências (Figura 7).

Figura 7 - Decompondo: soma por meio do padrão de Gauss

$$\begin{array}{r}
 \underline{295+6=301} \\
 \underline{296+5=301} \\
 \underline{297+4=301} \\
 \underline{298+3=301} \\
 \underline{299+2=301} \\
 300+1=301
 \end{array}$$

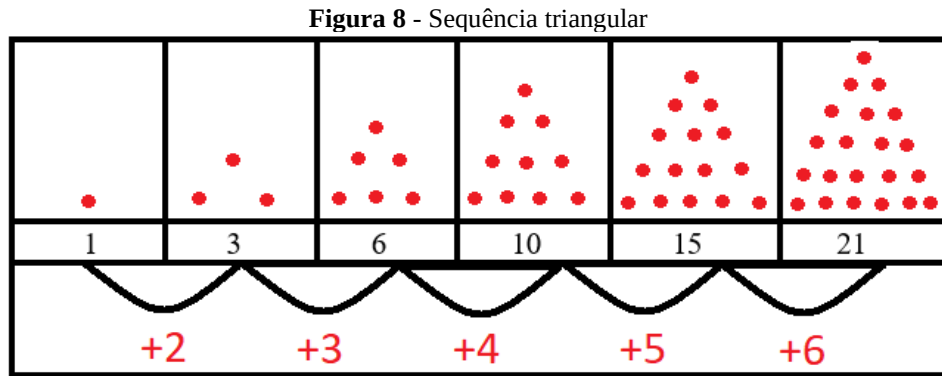
Fonte: desenvolvida pela autora (2020).

Quando dividimos o problema em equações menores e somamos o número de maior valor com o de menor valor sucessivamente, todos os resultados obtidos formam 301. Encontramos, pois, um padrão na equação.

Agora precisamos descobrir quantas vezes esse padrão se repete. Quantos pares iguais a esse conseguimos? Contar e responder, porém, levaria muito tempo, além de desgastar o aluno que está resolvendo o problema. Seria possível, então, dividir o maior número por 2, isto é, 300 dividido por 2: se a soma de cada par for 301, teremos então 150 pares, portanto $300/2 = 150$ pares. Dessa forma, podemos determinar a soma de todos os números.

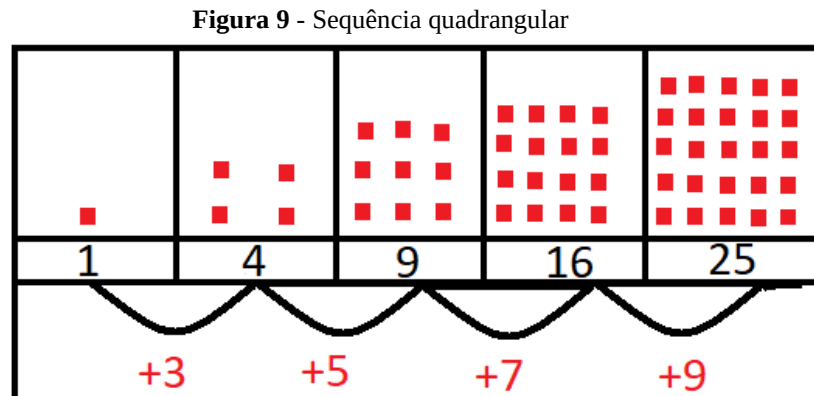
Outro exemplo em que podemos demonstrar o reconhecimento de padrões está presente na geometria. Chamaremos de Problema 4. Esse pode ser apresentado com o pensamento matemático por meio dos padrões numéricos e com agrupamento de pontos que formam figuras geométricas, como triângulos, quadrados ou pentágonos.

Observe a sequência triangular dos números — os quais podem ser chamados de números triangulares — 1, 3, 6, 10, 15 e 21 na Figura 8. Eles são encontrados por meio de um reconhecimento de padrão. No caso o padrão consiste em somar o valor da sequência anterior (+2+3+4+5+6...).



Fonte: desenvolvida pela autora (2020).

Temos ainda a formação dos números quadrangulares (Figura 9), entre os quais também podemos observar padrões. Os números quadrangulares são 1, 4, 9, 16, 25 e 36, e o padrão entre eles foi encontrado por meio da soma do número obtido na sequência com os números ímpares. Originam-se, assim, figuras quadrangulares, os quadrados perfeitos, os quais são números multiplicados por eles mesmos.



Fonte: desenvolvida pela autora (2020).

Embora exista um algoritmo em que um método possa ser mais rápido que outro, a correspondência de padrões é a chave para compor a solução.

2.5.5 Abstração⁶

Envolve o processo de verificar todos os fatores relevantes para um problema, podendo assim dar um enfoque maior ao que é realmente importante, sem que ocorra uma distração com dados desnecessários. Tudo que não é necessário é desprezado.

Para Wing (2008), a abstração é o principal elemento subjacente do PC, em que são recolhidas informações relevantes e inutilizados todos os dados irrelevantes. Para a autora, a abstração pode ser dividida em camadas: (1) abstração em cada camada; (2) abstração como um todo; (3) abstração entre camadas. Definindo-se as camadas, podemos relacioná-las, sabendo que é necessária a “abstração de um procedimento passo a passo para obter entrada e produzir a saída desejada” (WING, 2008).

Considerando o contexto do ensino de matemática, em situações simples ou complexas, podemos apontar dois exemplos de problema: (1) quando temos que resolver regras de fatoração; (2) quando nos deparamos com uma lâmpada queimada. Esses dois problemas têm aspectos diferentes. Para se solucionar o (1), é possível utilizar todo o conhecimento prévio do estudante sobre os conceitos matemáticos; já para o problema (2), seria necessário fazer diferentes perguntas para fazer a lâmpada funcionar, buscando diversas informações sobre sua condição, tais como o local onde está e como será possível removê-la. Ou seja, é preciso retornar à lâmpada, localizar o problema para que ela volte a funcionar. Com essas informações, torna-se possível descrever um algoritmo. Percebemos assim, a existência de várias camadas de abstração para resolução desse problema.

‘Abstrações’ são as ferramentas ‘mentais’ da computação. O poder de nossas ferramentas mentais é amplificado pelo poder de nossas ferramentas metálicas. Computar é a automação das nossas abstrações. Operação de mecanizar as nossas abstrações, camadas de abstração e seus relacionamentos. A mecanização é possível devido às nossas notações e modelos precisos e exigentes. Automatizar implica a necessidade de algum tipo de computador para interpretar as abstrações. (WING, 2008, p. 3.718).

Quando buscamos modelos para automatizar a resolução do problema, procuramos maneiras de interpretar as abstrações, as quais sugerem a identificação de semelhanças e diferenças entre problemas semelhantes, trabalhando em busca de uma solução para o problema original.

Sabendo disso, voltemos ao Problema 3, em que foi solicitada a soma dos números naturais entre 1 e 300. Nesse caso, para utilizar a habilidade da abstração, temos que buscar

⁶ Ação de abstrair, de analisar isoladamente um aspecto, contido num todo, sem ter em consideração sua relação com a realidade. (DICIO, 2022)

semelhanças e diferenças, focando apenas os detalhes mais importantes do problema. Vamos analisar a equação exposta no Quadro 8.

Quadro 8 - Soma em pares

$300+1 = 301$ soma dos pares	$300/2 = 150$ pares
------------------------------	---------------------

Fonte: desenvolvido pela autora (2020).

Podemos expressar o problema com a equação anterior. Se juntarmos essas equações, descobriremos a soma de todos os números entre 1 e 300 (Quadro 9).

Quadro 9 - Soma em pares com aplicação

$300+1 = 301$ soma dos pares	$300/2 = 150$ pares
$(300+1) * (300/2) = 45\ 150$	

Fonte: desenvolvido pela autora (2020).

Dessa forma, como focamos a atenção em detalhes, buscando as semelhanças e diferenças, a abstração acontece, quadro 9, e chegamos à resposta do problema.

2.5.6 Algoritmo

Afinal, o que é um algoritmo? De forma simplificada, trata-se de um processo de resolução de atividades ou processos por etapas. Cormen *et al.* (2009) definem algoritmo como uma sequência de passos computacionais que transforma dados de entrada em saídas.

Basicamente um algoritmo recebe dados, que são as informações a serem utilizadas para a resolução de um problema. Em seguida acontece o processamento, ou seja, o algoritmo precisa resolver o que se propôs. Por fim, há o procedimento de saída, que se caracteriza por exibir os resultados processados.

Dessa forma, um algoritmo se caracteriza pela entrada, pelo processamento e pela saída de dados (Figura 10). Notemos que, ao usarmos qualquer dispositivo eletrônico, esse processo é constante. Para cada comando solicitado, os *softwares* realizam a operação desejada e exibem um resultado. A figura abaixo representa as etapas de um algoritmo computacional de maneira gráfica.

Figura 10 - Fluxo de informações em um algoritmo



Fonte: desenvolvida pela autora (2022).

Por exemplo, seria possível elaborar um algoritmo que se propõe a somar valores digitados pelo usuário de forma indefinida, em que o usuário ou o próprio programa estabeleça um limite de valores a serem digitados. Após a resolução e prática desse algoritmo, podemos evoluir para um exemplo que utilize a fórmula de Gauss, levando em conta as estruturas de repetição que o algoritmo utilizará.

Com base na Figura 6, o algoritmo deve apresentar o resultado da soma de todos os números naturais de 1 até 300. Conforme Gauss notou, basta executar a multiplicação de 150×301 . Porém, a resolução algorítmica pode ser realizada por meio de dois laços de repetição: um primeiro valor inicial de entrada que começa em 1 e vai sendo acrescido de +1, enquanto, na outra ponta, um segundo valor de entrada recebe o valor inverso, ou seja, começa em 300 e vai subtraindo seu valor por -1. Uma terceira variável receberá continuamente a soma desses valores em um laço de repetição de 150 vezes, como demonstra a Figura 11.

Figura 11 - Algoritmo da fórmula de Gauss descrito em português

```

1 Algoritmo "FormulaGauss"
2
3 Var   inic, fim, soma: inteiro
4
5 Inicio
6     inic := 1
7     fim := 300
8     soma := 0
9
10    enquanto (inic <= fim) faca
11        soma := soma + (inic + fim)
12        inic := inic + 1
13        fim := fim - 1
14    fimenquanto
15
16    escreval(soma)
17
18 Fimalgoritmo

```

Fonte: desenvolvida pela autora (2022).

Observando o algoritmo em português da Figura 11, o estudante poderá desenvolver um aprendizado mais efetivo, afinal não aplicará puramente uma fórmula que nem sabe de onde surgiu, mas entenderá todo o raciocínio lógico para obtenção do resultado.

É importante ressaltar que, na resolução de algoritmos, os valores numéricos podem ser inteiros ou reais. Notemos que essa notação se inspira na teoria dos conjuntos da matemática,

porém de uma forma bem mais simplificada. No caso do exemplo do algoritmo da resolução de Gauss, os valores são apenas inteiros. Em uma resolução mais abrangente da RP por meio de algoritmos, podem ser necessários valores reais (com casas decimais), como no cálculo de média, por exemplo.

Ao considerar o PC como o processo de pensamento envolvido na formulação de problemas, podemos dizer que, a fim de suas soluções possam ser representadas como etapas e algoritmos computacionais, é importante encontrar modelos apropriados de cálculo com os quais o problema deverá ser formulado e derivar suas soluções para quaisquer categorias de problema.

Se considerarmos hoje um computador de baixo custo, certamente ele terá muito mais capacidade de resolução para cálculos do que a mente humana. Segundo a Lei de Moore⁷, a cada 18 meses, o poder computacional do computador duplica, ou seja, ele passa a ter muito mais capacidade de criar algoritmos precisos para resolver problemas. Na década de 1960, com a máquina de Turing⁸, era possível resolver problemas por meio de algoritmos bem mais simples do que temos hoje, porém ela ainda serve como modelo para resolução de problemas computacionais. Turing apresenta o algoritmo como uma descrição precisa para resolver problemas.

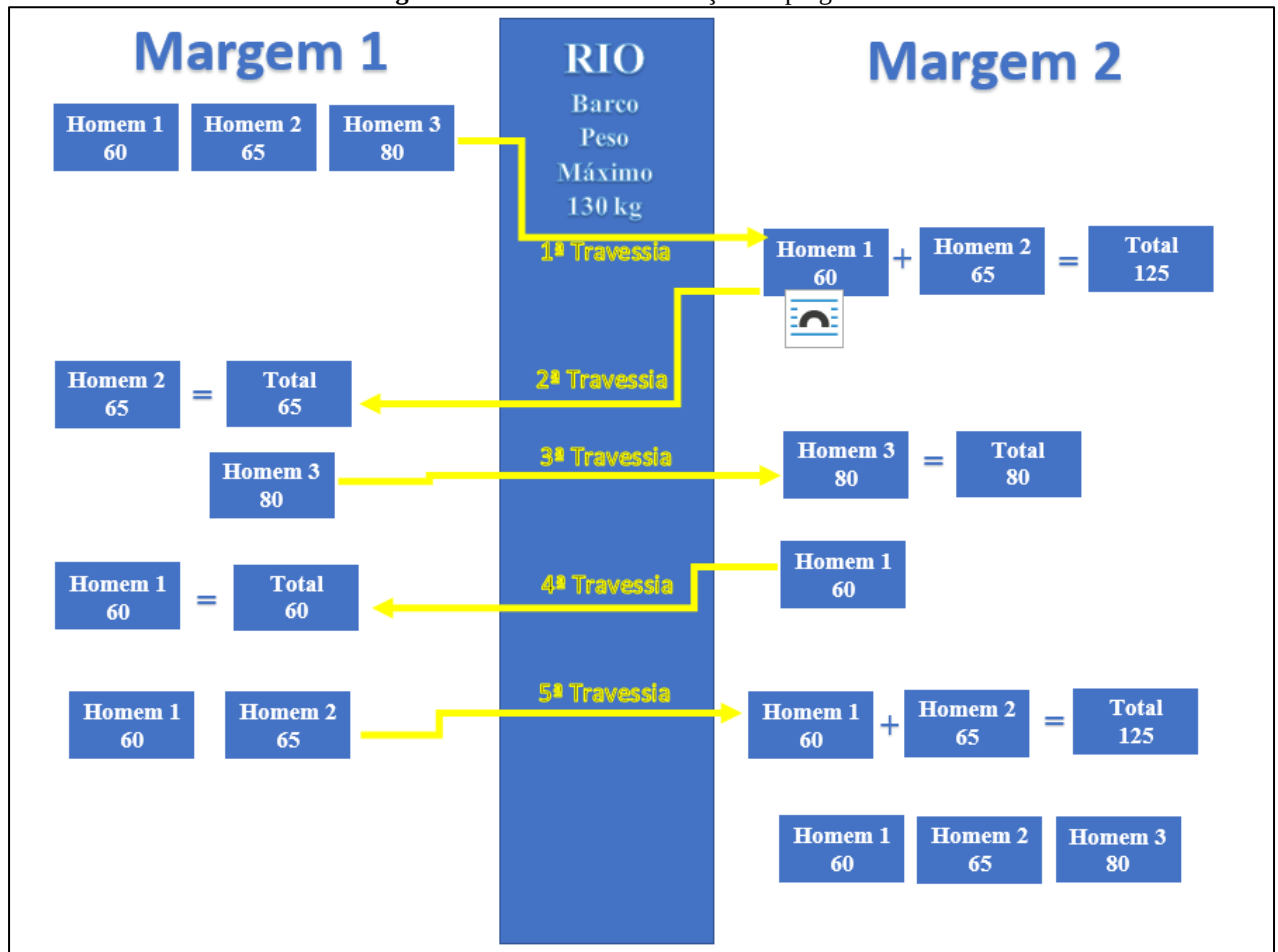
Para Wing (2006), o método cognitivo para construção de algoritmos para resolução de problemas é o que conhecemos por PC. O estudo de algoritmos envolve abstração, pois é necessário um refinamento nos conceitos.

Assim, podemos dizer que o algoritmo se caracteriza por constituir um conjunto de regras e de raciocínios lógicos empregados para a resolução de problemas de características semelhantes. Se observarmos o Problema 1, da travessia do rio, o algoritmo a ser descrito de forma desplugada (Figura 12) poderia ser:

⁷ A lei de Moore, de 1965, baseia-se na premissa de Gordon Earl Moore segundo a qual o poder de processamento dos computadores dobraria a cada 18 meses (MOORE et al., 1975).

⁸ A máquina de Turing é mais precisamente definida como um modelo abstrato de funcionamento de um computador, levando-se em conta apenas questões lógicas de seu funcionamento (TURING, 1936).

Figura 12 - Problema 1: resolução desplugada



Fonte: desenvolvida pela autora (2020).

Esse algoritmo poderia resolver outros problemas dessa mesma natureza, agora utilizando recursos tecnológicos. Outro exemplo seria o do problema da Figura 6: atrás da lógica do exercício, podemos demonstrar seu algoritmo também de maneira computacional, como desenvolvido a seguir por meio a linguagem *Processing* (Figura 13):

Figura 13 - Código-fonte em *Processing*

```

1  int x1 = 6;
2  int x2 = 295;
3
4  int soma = 0;
5
6
7  while ((x1 >= 1) && (x2 <= 300))
8  {
9      soma = x1 + x2;
10     println(x1, " + ", x2, " = ", soma);
11     x1 = x1 - 1;
12     x2 = x2 + 1;
13 }

```

Fonte: desenvolvida pela autora (2020).

O algoritmo acima resolve o que está proposto no item da decomposição, soma por meio do padrão de Gauss, e a programação foi implementada por meio da linguagem *Processing*. Primeiramente são declaradas as variáveis necessárias para a resolução do problema. O conceito de variável na programação é muito similar ao conceito do universo matemático, em que uma palavra ou letra é definida como uma variável.

A variável também precisa ter um tipo de dado, que pode ser numérico ou textual. No caso do algoritmo anterior, a variável é numérica e recebe apenas valores do conjunto dos números inteiros. Por isso, nas linhas 1, 2 e 4 do algoritmo, as variáveis são declaradas como inteiras — o tipo usado para essa declaração na maioria das linguagens de programação em inglês é “int” (abreviação de *integer*, que significa “inteiro” em português).

A partir da linha 7, é utilizada uma estrutura de repetição para dar conta da resolução do cálculo. Nesse exemplo, é empregado o *while* (“enquanto”, em inglês). Para serem executados os cálculos que fazem parte do enquanto, o teste deve ser verdadeiro. A partir do momento que o teste é falso, o laço de repetição é interrompido.

Dentro da estrutura de repetição, acontece a soma das variáveis *x1* e *x2* (linha 9), que são armazenadas em uma variável chamada justamente de *soma*. Na linha 10, o resultado é exibido para o usuário, e na sequência é realizado um decremento (subtração de -1) na variável *x1* e um incremento (adição de +1) na variável *x2*.

Assim, o laço de repetição vai automaticamente realizando cálculos até o momento em que o teste se torne falso — nesse caso, quando *x1* for menor que 1, e *x2*, maior que 300.

Ao criarmos um modelo computadorizado como o algoritmo acima, adicionando recursos da linguagem de programação *Processing*, temos diferentes conceitos matemáticos lógicos

interligados. Sabemos que a abstração matemática não usufrui de propriedades algébricas claras, elegantes e facilmente identificáveis (KONG; ABELSON, 2019).

Por fim, ao estudarmos essas quatro habilidades do PC (Figura 14), conseguimos criar um escopo simplificado para maior compreensão de como o PC está sistematizado e de como levar os estudantes a pensar computacionalmente ao resolver problemas matemáticos.

Figura 14 - Habilidades do PC



Fonte: desenvolvida pela autora (2020).

Finalmente, temos a decomposição como divisão do problema em unidades menores, o reconhecimento de padrões como verificação de tudo que temos em comum dentro do problema, a abstração para modelar aspectos principais do problema, e o algoritmo, que registrará todo o plano de instruções.

2.6 Educação matemática e PC

Considerando que, atualmente, percebemos “um baixo rendimento na aprendizagem de Matemática pelos alunos no ensino básico” (BARCELOS; SILVEIRA, 2015), é necessário pensar em soluções que lidem com esse problema de maneira efetiva. Assim, levando em conta as questões teóricas elencadas anteriormente neste capítulo, na conceituação tanto da RP como

do PC, entendemos que ambos os saberes se interligam, e o conjunto de competências que podem ser aprimoradas com o PC na educação matemática são diversas. Vamos a alguns exemplos:

- Um problema resolvido pelo PC é solucionado de forma mais simples, pois presume que “isso exige a capacidade de pensar de forma abstrata e em múltiplos níveis, e não a mera aplicação de técnicas de programação” (WING, 2006).
- O PC proporciona ferramentas como os computadores para resolver os problemas, de modo que a realidade social pode ser amplificada por conhecimentos de diferentes áreas, tais como matemática e ciências, que exigem um maior rigor lógico.
- A utilização do computador para resolver problemas potencializa a forma de pensar e propõe ideias para a resolução, utilizando o processamento informacional do computador.
- O PC precisa estar dentro de uma metodologia ativa⁹ na educação matemática, auxiliando o aluno a estudar, opinar e ser crítico à forma de solucionar.
- O PC abre um leque para diferentes áreas da educação, desde que haja uma resolução adequada. Estas áreas estão principalmente relacionadas às ciências exatas, como física, química, matemática e ciências, por exemplo.

O envolvimento do PC com a Educação Básica pode resultar no desenvolvimento de habilidades que “auxiliam na resolução de problemas, dando suporte e se relacionando com outras ciências, além de atuar como um elemento motivador para os estudantes”. “Incorporar o pensamento computacional à Educação Básica envolve a análise sistemática de seu potencial e a sinergia com outras áreas do conhecimento, como a Matemática” (BARCELOS; SILVEIRA, 2012).

Aplicar o PC na matemática se torna benéfico (e até mesmo necessário) porque “os alunos que precisam desenvolver habilidades que vão além dos cálculos, eles precisam ter a capacidade de descrever, explicar ou prever fenômenos, utilizando várias formas de raciocínio” (OCDE, 2014 apud MESTRE *et al.*, 2015). Assim, “o processo de ensino-aprendizagem de Matemática seria beneficiado pela incorporação do Pensamento Computacional ao currículo do ensino básico” (BARCELOS *et al.*, 2015).

Wing (2008) postula que o conceito de computação inclui resolução de problemas e comportamento humano. Para a autora, computar é pensar; é ver, de maneira funcional, a forma como o problema é resolvido. Nesse sentido, a autora acredita que a essência do PC está diretamente ligada à matemática. Por isso, muitos pesquisadores educacionais estão trabalhando

⁹ Em que o aluno é protagonista do seu aprendizado e é estimulado a buscar respostas aos problemas. Nessa proposta o docente é um facilitador do conhecimento, que é construído pelo próprio estudante.

para integrar no currículo esse componente, focando computação, ciência, tecnologia e matemática (KONG; ABELSON, 2019).

Em seu livro *Computational Thinking Education*, Kong e Abelson (2019) designam o termo “computar” conforme Seymour A. Papert. Os autores realizam um estudo em uma escola primária onde abordam a criação de um aplicativo com a finalidade de aprofundar conceitos matemáticos, como aprender números compostos e primos, e ainda desenvolver habilidades e perspectivas do PC, assim os alunos conseguem experiência operacional algorítmica e concreta. Eles chamam essa pedagogia de inovadora, pois consegue fazer o estudante desenvolver habilidades lógicas e computacionais.

A implementação do PC nas escolas pode trazer vários benefícios para a formação dos estudantes da atualidade. Sabemos que se discutem alguns desafios a serem enfrentados na busca pelo seu ensino na Educação Básica no Brasil (FRANÇA; TESDECO, 2016), mas acredita-se também que pode contribuir para a inclusão digital nas escolas. O PC, em conjunto com a matemática, promove o ensino-aprendizagem ao proporcionar conteúdos que favoreçam a construção do conhecimento.

Para se promover uma aprendizagem significativa fazendo uso do PC, autores como Lee (2011) apontam que é necessário acesso a ambientes ricos de aprendizagem que incluam professores qualificados e, se possível, a inserção de novas tecnologias. No entanto, o campo requer procedimentos sistemáticos de avaliação baseados em pesquisa das ciências da aprendizagem para descrever a progressão e o desenvolvimento das habilidades do PC na educação.

E é para descrever como acontece o processo de desenvolvimento de RP e como esse processo pode ser melhorado que uniremos habilidades matemáticas a computacionais.

O uso do PC e de programação é referência em escolas de diversos países, tanto de Ensino Fundamental como de Médio. É também referência em várias literaturas que investigam as habilidades do PC misturando lógica, raciocínio e competências computacionais que habilitam o estudante a pensar de maneira autônoma (KONG; ABELSON, 2019). Possibilita também o desenvolvimento de habilidades de codificação em crianças menores, englobando aspectos de sequenciamento, reconhecimento de padrões, raciocínio dedutivo, raciocínio lógico, estruturas e representações de dados e funções (LIUKAS, 2015), determinado assim ligações importantes e fundamentais entre codificação precoce e aprendizagem matemática.

No Brasil, introduzir o PC na Educação Básica é algo recente, aprovado em 2018 com a BNCC (Base Nacional Curricular Comum). O documento preconiza que todos os estudantes devem desenvolvê-lo ao longo dos anos escolares. Ao se fazer um estudo da BNCC, observa-se

o uso do termo “Pensamento Computacional” nove vezes, fazendo-nos refletir sobre a necessidade de um estudo voltado a esse componente que pode trazer diferentes benefícios ao ensino. A BNCC propõe que as habilidades do PC devem ser trabalhadas desde os anos iniciais do Ensino Fundamental, de forma que, ao chegar aos anos finais, o estudante consiga interpretar e elaborar algoritmos por fluxograma. Já no Ensino Médio, dá-se continuidade, por meio do conhecimento adquirido, à resolução de problemas mais complexos que necessitam de maior análise, criticidade e abstração (BRASIL, 2018, p. 471).

A abordagem presente na BNCC para o PC está relacionada à disciplina de matemática, que articula a grade em aritmética, álgebra, geometria, estatística e probabilidade, visando a preparar o aluno para resolver problemas de qualquer natureza, dentro e fora dos espaços escolares, onde essas situações sigam como referência a vivência diária do aluno em seu cotidiano e considerando que essa resolução aconteça por meio dos algoritmos (BRASIL, 2018, p. 271).

[...] as competências e habilidades de raciocinar, representar, comunicar e argumentar matematicamente, de modo a favorecer o estabelecimento de conjecturas, a formulação e a resolução de problemas em uma variedade de contextos, utilizando conceitos, procedimentos, fatos e ferramentas matemáticas. (BRASIL, 2018, p. 264).

Essa revisão do currículo vem, como em uma de suas principais vertentes, adicionar ambientes de aprendizagem integrados que permitam aos alunos fazer uso de dimensões do PC para o desenvolvimento de habilidades na RP. A necessidade de que os alunos incorporem, de maneira mais assertiva, a RP tornou-se um tema relevante em muitos padrões internacionais, como o PISA.

Esses processos de aprendizagem são potencialmente ricos para o desenvolvimento de competências fundamentais para o letramento matemático (raciocínio, representação, comunicação e argumentação) e para o desenvolvimento do pensamento computacional. (BRASIL, 2018, p. 264).

Considerando essa realidade e conhecendo a organização da nossa educação no que se refere à disciplina de matemática, partiremos do ponto de que a RP é uma habilidade primordial aos estudantes. Tornou-se o meio para reunir conteúdo e aplicações em ambientes escolares, para que se possa interpretar informações e interagir na sociedade atual, além de ser aplicável em diversas áreas do conhecimento. Ademais, o método e o processo desenvolvidos durante a RP são fundamentais para alcançar um bom desempenho e adquirir habilidades para, inclusive, melhor utilizá-la.

Embora muitos alunos não possuam as habilidades básicas para resolução de problemas simples ou complexos, todo campo de trabalho exige alto nível de conjunto de habilidades, já que as novas tecnologias mudaram a natureza da maioria de todos os locais de trabalho.

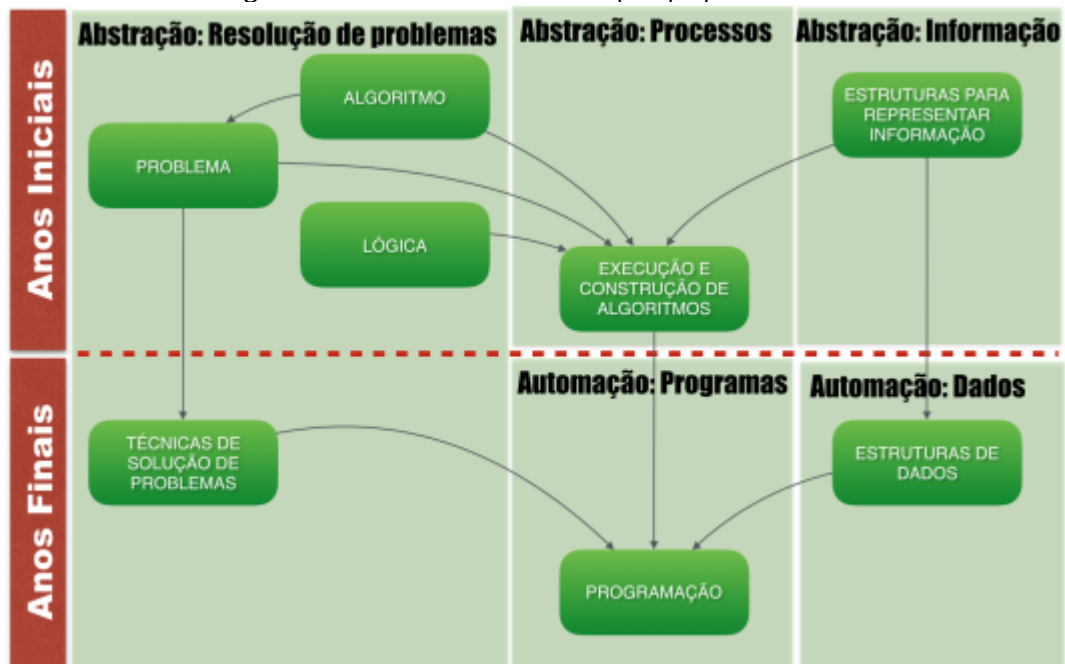
Sob a influência das tecnologias, o olhar para a RP vem sendo revisto, passando a ser enxergado como uma atividade mental complexa que exige uma variedade de habilidades e ações cognitivas para o seu bom desenvolvimento. Essas habilidades incluem visualização, associação, abstração, compreensão, manipulação, raciocínio, análise, síntese e muitas outras que jamais pensamos estar sendo utilizadas. Os profissionais da área precisam saber como esse processo pode ser tratado de formas diferenciadas, que corroborem para sua melhor aplicabilidade.

Ao aprender RP, os estudantes passam a adquirir a habilidade e a capacidade de conectar ideias matemáticas e desenvolver a compreensão de conceitos. Esse processo necessário e essencial aproxima a matemática de suas práticas cotidianas, justamente por ser mais contextualizado e efetivo. A tecnologia pode ser uma aliada nessa contextualização.

Em seu estudo, Marchisio *et al.* (2022) afirmaram que, ao abordar a RP dentro de um ambiente de aprendizagem digital e tecnológico, em conjunto com outras metodologias, permitiu-se que o aluno aprendesse melhor matemática, mesmo não tendo muita facilidade com a disciplina. Verificou-se, assim, que a aprendizagem contextualizada com a RP e com recursos tecnológicos facilitou a aprendizagem.

Usar programação e desenvolver habilidades de PC como intervenção pedagógica na Educação Básica é uma proposta também da Sociedade Brasileira de Computação (SBC), que traz, em sua Diretriz para Ensino de Computação na Educação Básica, a importância de se introduzir conceitos e o desenvolvimento de habilidades do PC na educação, como mostra a Figura 15.

Figura 15 - Eixos do PC articulados pela proposta da SBC



Fonte: eixo do PC no Ensino Fundamental, proposta da SBC (RAABE, 2017, p. 8)

Ao analisarmos o processo de ensino e aprendizagem dos conhecimentos matemáticos nos anos iniciais, percebemos que ele é dinâmico, afinal as atividades envolvem investigação, ou abstração. Constata-se a busca por respostas, saídas com formulação de hipótese e o que proporciona a base para a criação estratégia para RP fazendo uso do eixo do PC (Figura 15).

Devem dominar as principais operações para a construção de algoritmos (composição sequencial, seleção e repetição) e ter noções de técnicas de decomposição de problemas. (BRASIL, 2021, p. 29).

Atualmente, o ensino de aritmética nos anos iniciais tem como base o pressuposto de que a matemática é um conjunto de regras e símbolos para o aluno internalizar. No entanto, a teoria da Jean Piaget (1896-1980) chamada construtivismo¹⁰ mostrou que jovens e crianças adquirem conhecimento matemático por meio da interação com o meio ambiente. Nessa etapa inicial, é extremamente importante que os conceitos sejam introduzidos sempre por meio de experiências concretas, que serão fundamentais para a formação de um protótipo mental para abstrações computacionais, as quais servirão de base para atividades futuras de programação.

¹⁰ Trata-se de uma teoria que acredita que, para o desenvolvimento da aprendizagem acontecer, a criança passa por quatro estágios, nos quais o conhecimento cognitivo é produto de uma construção pessoal do aluno.

Nesse sentido, este trabalho toma como base atividades de programação que chamaremos de *desplugadas*¹¹ com a utilização do programa *Processing*. Como não tínhamos conhecimento da etapa de desenvolvimento em que se encontravam os participantes da pesquisa em relação às habilidades do PC, trabalhamos com essas atividades desplugadas, que deveriam ser introduzidas nos anos iniciais, para que pudéssemos entender todo o processo de RP desses participantes sem negligenciar nenhuma etapa.

Segundo a SBC, para os estudantes de anos finais do Ensino Fundamental, exige-se a aptidão para realizar atividades com diferentes técnicas de construções algorítmicas, realizar descrição das soluções, construir modelos computacionais de sistemas simples ou complexos, argumentar criticamente a respeito de todo processo de solução de problemas, além de ter todo conhecimento sugerido pela Competência Digital sobre cuidados e limite no uso da internet.

2.7 O ambiente de programação do *Processing*

Na literatura, verificam-se diferentes estudos sobre o uso do PC. Como já citado neste trabalho, entre os resultados e as conclusões mais frequentemente apresentados, estão atividades com a utilização do Scratch em primeiro lugar e, em seguida, atividades desplugadas.

O presente trabalho utiliza um ambiente de programação diferente, o *Processing*, para abordar o uso do PC. Para que possamos entender como esse ambiente funciona e onde se passa a pesquisa dentro desse referencial teórico, explanaremos como esse ambiente funciona, suas principais características e usabilidade por meio de algumas aplicações, para que fique claro como essa ferramenta pode auxiliar nesse estudo.

2.7.1 Breve resumo das origens do *Processing*

O *Processing* é um ambiente de programação lançado no ano de 2001 por dois estudantes do MIT, Ben Fry e Casey Reas (PROCESSING, 2021). Tal ambiente possui código-fonte aberto e é voltado para iniciantes no aprendizado de lógica de programação e algoritmos. O programa pode ser acessado via *site* oficial (Figura 16) por meio do *link* Processing.org.

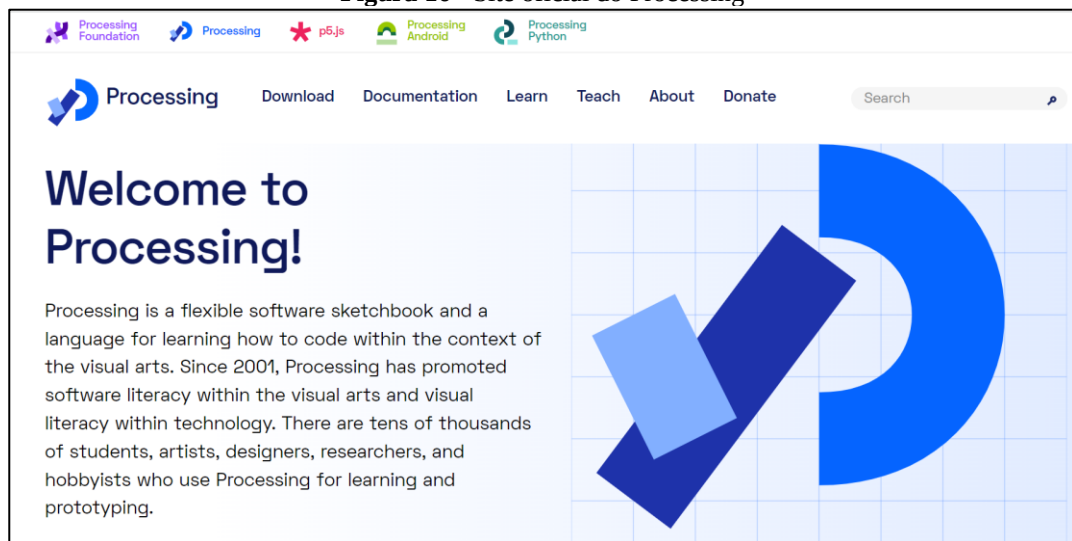
Como relatado em seu *site* oficial, na seção Overview, o *Processing* já possui 20 anos de existência e se tornou uma alternativa de aprendizado para iniciantes em programação, pois muitos outros ambientes e linguagens de programação são proprietários. Por esse motivo,

¹¹ Desplugadas são atividades para desenvolver lógica de programação ou atividades voltadas à computação sem a necessidade de um computador.

acarretam um custo que a maioria daqueles que gostaria de aprender uma linguagem não consegue arcar (PROCESSING, 2021).

Outro ponto importante é a experiência do uso do *Processing* nos últimos anos, pois instituições de ensino de vários níveis vêm o aplicando com sucesso, inclusive integrando-o com plataformas como o Arduino¹² em cursos universitários. Os museus de São Francisco, Nova York e Londres firmaram parcerias para aplicação do ambiente em seus projetos artísticos.

Figura 16 - Site oficial do Processing



Fonte: Processing (2021).

Como o código-fonte do *Processing* é código-aberto, o ambiente é dependente de colaboradores e doadores. No *site* oficial, existe a possibilidade de doação para o projeto. Também é possível consultar a documentação, fazer o *download* de versões para *desktop*, aprender com exemplos práticos e obter outras informações sobre o ambiente.

2.7.2 O ambiente Processing, derivados e evoluções

Como em qualquer ramo científico, a dita ciência da computação começou na primeira metade do século passado, com equipamentos eletrônicos enormes se comparados aos dispositivos atuais e com pouquíssima mão de obra para programá-los. Geralmente eram os cientistas que lideravam grandes grupos de pesquisadores que realizavam esse serviço.

¹² É mais adequadamente definido como um sistema de *hardware* e *software* integrado que pode se conectar a um computador e que facilita que as pessoas consigam desenvolver projetos e colocar em prática suas ideias (MCROBERTS, 2018).

Aos poucos tanto equipamentos quanto linguagens e ambientes de programação foram se modernizando, de modo que alcançamos o cenário atual, em que aprender programar tornou-se uma habilidade quase interdisciplinar, pois cursos como jornalismo e direito estão ensinando programação para seus estudantes. Mas como isso é possível?

Nos últimos anos, estamos deixando o paradigma de linguagens mais generalistas, como a C, PHP ou Java, para lidar com linguagens e ambientes de programação de cunho — isto é, finalidade — mais específico, como Python, R e *Processing*.

Contudo, vários grupos de pesquisa internacionais em grandes universidades desenvolveram linguagens e ambientes de programação voltados ao desenvolvimento de aplicações e que também possuem características para ajudar os estudantes a entender melhor os conceitos, muitas vezes, áridos da programação. Por exemplo, Papert (1986) criou o ambiente de programação logo, que buscava estabelecer conexões entre os conceitos do construtivismo de Jean Piaget com ainda incipiente formação de programadores nas escolas de Ensino Fundamental e Médio. O autor classificou o conceito como construcionismo para conectar as ideias. Outra iniciativa bem-sucedida foi do MIT, o Scratch, que ensina conceitos de programação por meio de “blocos”, em que se monta um algoritmo como se fosse um quebra-cabeças.

Nesse contexto também se encaixa o *Processing*, pois ele foi desenvolvido a partir de uma linguagem amplamente utilizada, com foco mais específico. Assim como os outros ambientes anteriormente citados, pode se vincular ao ensino em seus diversos níveis. No Apêndice 1 deste trabalho (Iniciativas Nacionais e Internacionais sobre Pensamento Computacional) são apresentados outros projetos que têm similar vocação ao *Processing*.

2.7.3 Bases do ambiente de programação do *Processing*

Outro aspecto importante do *Processing* é que seu ambiente de programação se baseia na linguagem de programação Java e já vem em um pacote que inclui o editor de textos para o código-fonte do *software* e compilador (*software* encarregado de traduzir o código-fonte para uma aplicação possível de ser executado pelo computador).

Nos últimos anos, o *Processing* também ganhou uma versão *online*, disponível em <https://openProcessing.org/>, bastando assim uma conexão com a internet para que o estudante possa resolver os algoritmos por meio da programação em um navegador Web, como Google Chrome, por exemplo (OPEN PROCESSING, 2021).

É importante ressaltar aqui que a versão *online* é mais limitada do que a *desktop* (instalada no próprio computador). Para o início do aprendizado, a primeira funciona bem, porém, conforme há mais sofisticação do código, torna-se necessário migrar para a segunda¹.

Como uma evolução natural do *Processing*, ele vem se integrando com outros ambientes e linguagens, como Android, programação para dispositivos móveis (PROCESSING FOR ANDROID, 2021) e também Python, utilizado para a área já bem-estabelecida de ciência de dados (PYTHON MODE FOR PROCESSING, 2021).

Como ocorre com a maioria das linguagens e dos ambientes de programação, o resultado dos algoritmos realizados em *Processing* é imediato, ou seja, ao fazer um cálculo seguindo as etapas estipuladas de construção do algoritmo, o aprendiz da RP por meio de programação já verifica se o que codificou está correto ou se necessita de ajustes em tempo real.

Além disso, o *Processing* possui também um IDE (*Integrated Development Environment* – Ambiente de Desenvolvimento Integrado), para que os estudantes da linguagem possam rapidamente desenvolver maior familiaridade com seus códigos, bem como um aprendizado mais veloz. O IDE está disponível para os sistemas operacionais Windows, Linux, Mac OS (sistema operacional dos equipamentos *desktop* da Apple) e na versão Web.

O *Processing* pode ser comparado a um dialeto que tem sua sintaxe baseada em Java, que, por sua vez, também possui elementos da flexível e amplamente utilizada linguagem C (REAS; FRY, 2015). Além disso, possui diversas bibliotecas de funções para desenhos gráficos. É possível desenhar e movimentar objetos quando da compilação do código-fonte de um programa. Com isso, os estudantes podem entender, de maneira simples, como uma animação é feita e como pode ser complexa a programação por trás de diversas animações a que assistimos em nossos momentos de lazer em serviços de *streaming* ou no cinema, por exemplo. Essa facilidade de realizar animações é baseada em uma estrutura de rotinas já previamente definidas, dessa forma facilitando a codificação.

Por possuir diversas funções voltadas a *design* e animações, o ambiente do *Processing* vem ganhando cada vez mais espaço em escolas de arte, como forma de ensinar programação para designers digitais e artistas em geral. Venturelli (2010) argumenta que a arte está cada vez mais digital, portanto, se torna importante que profissionais da área de *design* e artistas em geral aprendam programar, pois se trata de uma nova demanda importante para o desenvolvimento profissional da área.

É importante ressaltar, de forma alinhada aqui com os objetivos do presente trabalho, que esse aspecto mais voltado a resultados visuais do *Processing* representa para os estudantes resultados mais expressivos (PROCESSING, 2021).

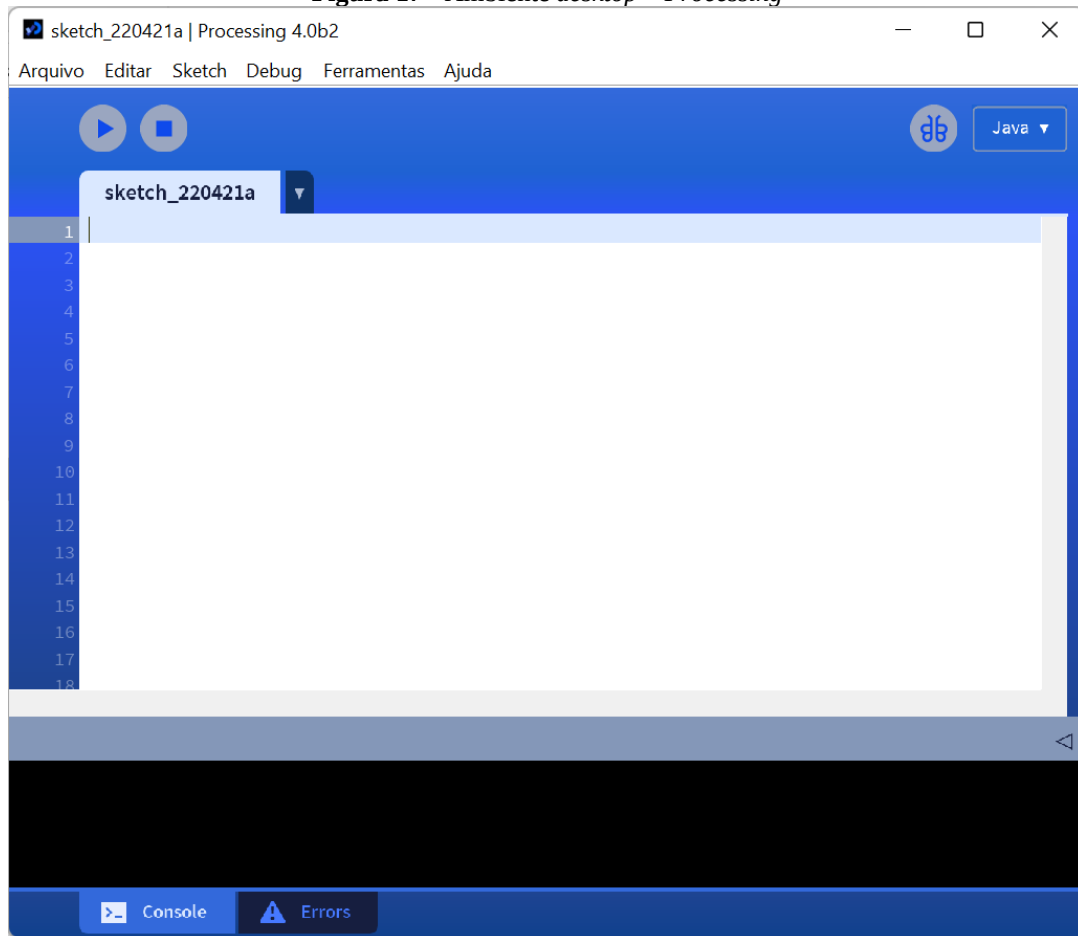
No trabalho de Souza (2018), o *Processing* é empregado como apoio, na disciplina de matemática, para alunos do Ensino Fundamental com baixo rendimento. O autor ressalta que a utilização da programação possibilitou identificar estudantes com talento na área, os quais, porém, “são ignorados ou taxados de ruins”, por não obterem rendimento satisfatório em matemática.

2.7.4 Ambiente desktop do Processing e suas funcionalidades

Em seus mais de 20 anos, o ambiente do *Processing* proporcionou aos usuários e novos aprendizes diversas modificações. Atualmente, a última versão é a 4.0 beta 7 (*PROCESSING*, 2021). Como visualizado na Figura 18, o IDE de desenvolvimento é bem simplificado e apresenta poucos botões e menus. Porém, a linguagem possui muitas bibliotecas, documentação e funções que facilitam o desenvolvimento de programas complexos com poucas linhas de programação.

Além disso, como já citado, o ambiente vem obtendo integrações com outras linguagens e ambientes de programação, o que confere mais poder de desenvolvimento para os usuários interessados em começar a aprender a programar, bem como para usuários com finalidades mais específicas, como *designers* e artistas que usam aplicações digitais.

Sobre o ambiente apresentado na Figura 17, verificam-se apenas seis menus, incluído o menu Ajuda, que existe em todos os *softwares*. Logo abaixo, há somente dois botões, o primeiro para rodar a aplicação e o segundo para caso necessário, interromper a execução da aplicação.

Figura 17 - Ambiente *desktop* – Processing

Fonte: desenvolvida pela autora (2021).

A seguir apresentamos um exemplo de programação e posterior explicação do código-fonte desenvolvido no próprio IDE do *Processing*. O código faz com que um círculo branco se movimente do lado esquerdo para o direito de maneira constante (Figura 16). Analisando esse código, constatamos que o *Processing* possui duas importantes funções no desenvolvimento de aplicações: `void setup ()` e `void draw()`. A função `void setup ()` é executada uma única vez.

No código-fonte a seguir, há três comandos que pertencem a essa categoria. Na linha 3, é definido o tamanho da tela por meio da função `size()`, que possui dois parâmetros numéricos em que se demarca a largura e a altura em *pixels* da tela produzida pela aplicação. Na linha 4, a função `background ()` define a cor de fundo de tela utilizando o consolidado modelo RGB (*Red*, *Green*, *Blue*), em que os valores de vermelho, verde e azul variam de 0 a 255. Por fim, na linha 5, o `frameRate()` define a quantidade de vezes em que o código na função `draw()` será executado por segundo; quanto maior o valor do `frameRate()`, mais acelerada ficará a animação. A Figura 18 apresenta o código-fonte explicado acima e, a partir da sua execução, a animação ocorrerá (Figura 19).

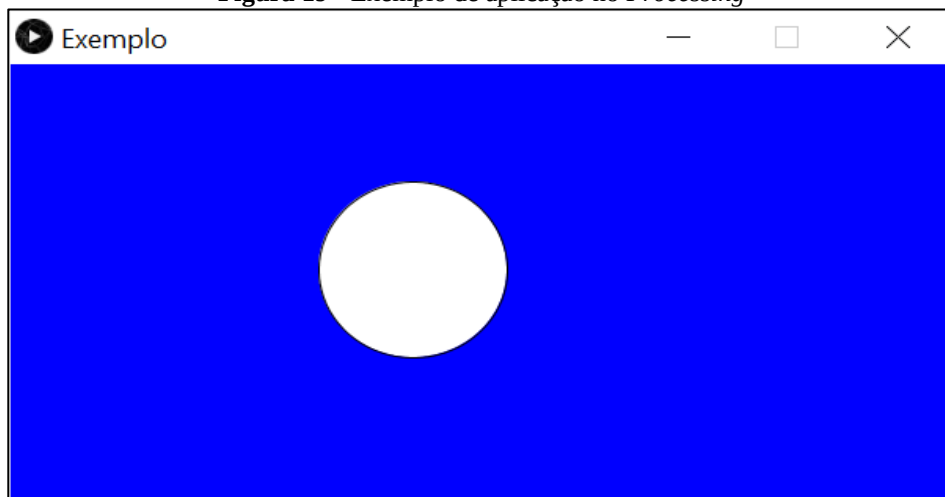
Figura 18 - Código-fonte na linguagem *Processing*

```

Exemplo ▼
1 void setup()
2 {
3     size(600,300);
4     background(0,0,255);
5     frameRate(200);
6 }
7
8 int x = 60;
9 int y = 140;
10 String idavolta = "ida";
11
12 void draw()
13 {
14     background(0,0,255);
15
16     if ((x <= 550) && (idavolta == "ida"))
17     {
18         ellipse(x, y, 120, 120);
19         x = x+1;
20         if (x == 540)
21         {
22             idavolta = "volta";
23         }
24     }
25     else
26     {
27         ellipse(x, y, 120, 120);
28         x = x-1;
29         if (x == 60)
30         {
31             idavolta = "ida";
32         }
33     }
34 }

```

Fonte: desenvolvida pela autora (2020).

Figura 19 - Exemplo de aplicação no *Processing*

Fonte: desenvolvida pela autora (2020).

2.7.5 Utilizando o Processing na resolução de problemas matemáticos

No contexto do objeto de estudo desta pesquisa, o *Processing* pode ser usado na resolução de problemas matemáticos que visem a apresentar os resultados de fórmulas e cálculos matemáticos utilizando uma linguagem de programação e a capacidade de processamento de algum dispositivo eletrônico.

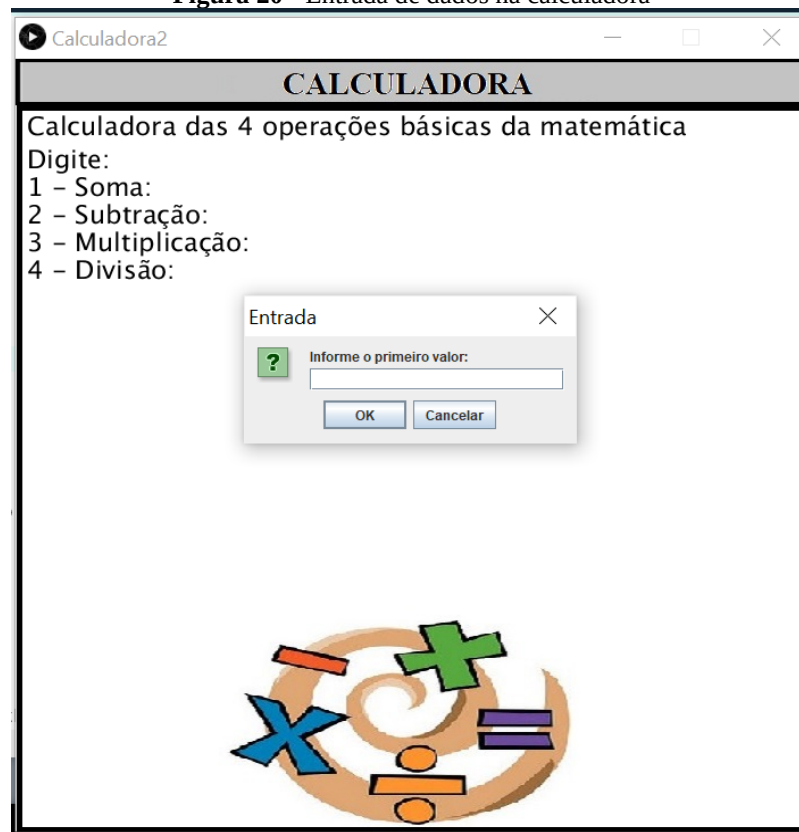
Por exemplo, podemos utilizar o *Processing* para criar algoritmos que resolvam cálculos matemáticos básicos, desde estruturas simples, como as quatro operações básicas (soma, subtração, multiplicação e divisão) em uma aplicação que simula uma calculadora, até mesmo para cálculos considerados mais complexos pelos estudantes, como equações de segundo grau. No próximo tópico, serão apresentados alguns programas feitos em *Processing* que realizam cálculos matemáticos básicos.

2.7.6 Exemplos de utilização do Processing em conteúdos matemáticos

Neste tópico será observado como o uso do ambiente de programação *Processing* pode ser utilizado na disciplina de matemática como um recurso pedagógico e como suas propriedades podem auxiliar na resolução de problemas simples ou complexos. Ao mesmo tempo que o estudante aprende por meio da construção do algoritmo, nota que os conteúdos da matemática se encaixam muito bem dentro dessa proposta, o que pode gerar um ganho adicional no aprendizado da disciplina.

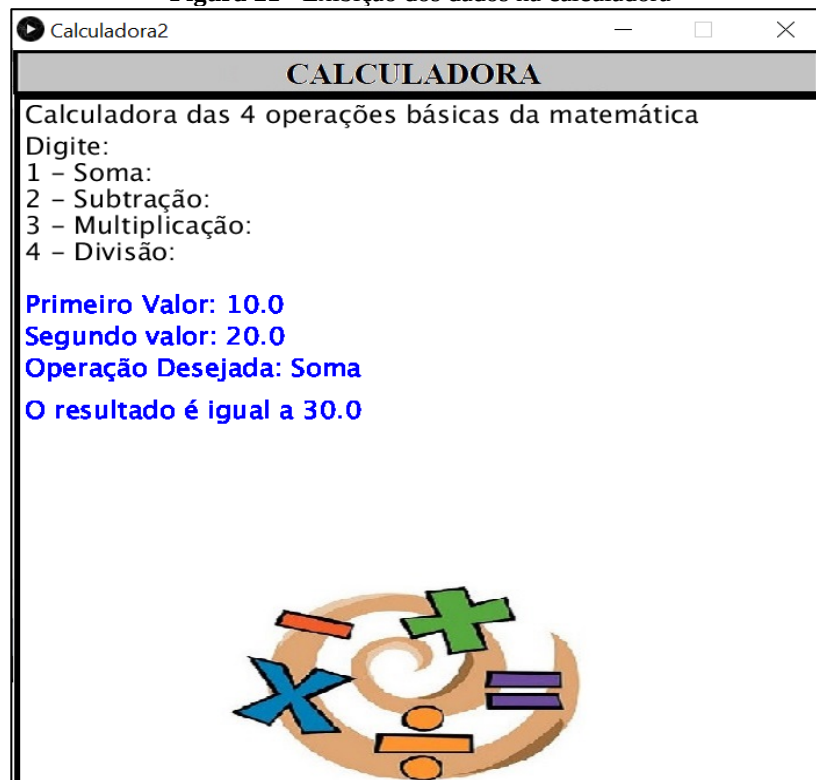
O primeiro exemplo é o da calculadora das quatro operações básicas da matemática, como demonstrado na Figura 20. O estudante digita os valores numéricos que deseja, escolhe a operação desejada, e o programa desenvolvido em *Processing* retorna o resultado da operação selecionada, como mostra a Figura 21. Seguem abaixo duas telas que detalham essa explicação:

Figura 20 - Entrada de dados na calculadora



Fonte: desenvolvida pela autora (2020).

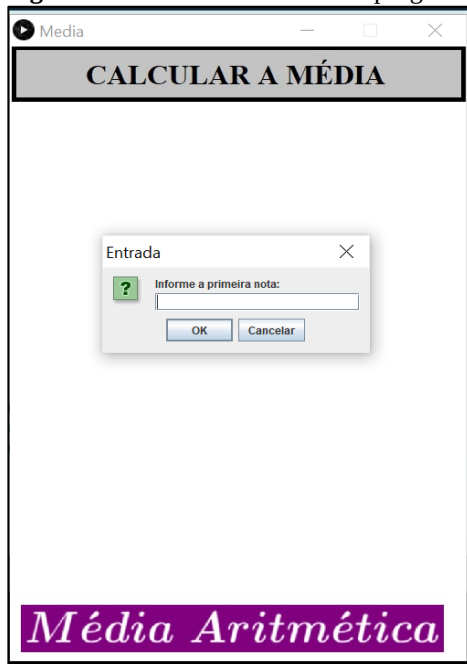
Figura 21 - Exibição dos dados na calculadora



Fonte: desenvolvida pela autora (2020).

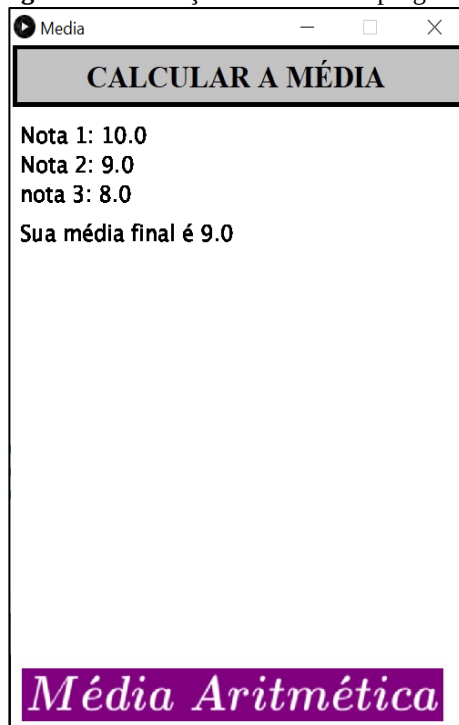
O segundo exemplo envolve o cálculo da média aritmética de três notas digitadas pelo estudante. Após a entrada dos dados, o algoritmo inserido no programa em *Processing* calcula a média e exibe o resultado. As Figuras 22 e 23 exemplificam a entrada dos valores das notas e a posterior exibição do resultado da média.

Figura 22 - Entrada dos dados no programa



Fonte: desenvolvida pela autora (2020).

Figura 23 - Exibição dos dados no programa



Fonte: desenvolvida pela autora (2020).

2.8 Atividades desplugadas

Uma das premissas do PC consiste na ideia de que o conhecimento de computação vai muito além do usuário de dispositivos tecnológicos. Somente a inserção de tecnologia em sala de aula não confere maior aprendizado em qualquer área. Considerando isso, o PC objetiva possibilitar conhecimentos sobre computação que contribuam para o desenvolvimento do conhecimento científico e potencializando a capacidade de solução de problemas e posterior criação de produtos e processos (RAABE *et al.*, 2017).

Nesse contexto, as atividades desplugadas — ou seja, sem a utilização de dispositivos eletrônicos — são maneiras de desenvolver as habilidades do PC, como algoritmo, por exemplo, de forma mais natural e agregando conhecimento que será útil para o aprendizado da matemática.

Em trabalho realizado por Taub, Armoni e Ben-Ari. (2012), os estudantes participantes consideram que os profissionais da computação necessitam do conhecimento matemático como ferramenta para o estudo da área, porém a pesquisa não identifica mais a fundo a natureza dessas relações que os estudantes intuem.

Sem dúvida a matemática é um pilar importante para o desenvolvimento do conhecimento de diversas áreas, inclusive atualmente é difícil encontrar um campo que não use as ferramentas matemáticas para o desenvolvimento de pesquisas científicas. Assim, as atividades desplugadas conferem um ponto de conexão entre atributos do PC e a matemática, de forma que os estudantes conseguem construir um conhecimento mais contextualizado e sólido a partir dessas premissas.

Autores como Machado *et al.* (2010) definem atividade desplugada como uma técnica cujo objetivo é aprimorar os conhecimentos da ciência da computação por meio da ludicidade, sem uso de computadores, podendo qualquer pessoa de qualquer idade participar delas, justamente por serem simples.

Com o avanço da tecnologia e da computação na sociedade, as disciplinas escolares passam a cada vez mais se entrelaçarem com a computação. A fim de preparar melhor os alunos para a alfabetização computacional, novos cursos vêm ganhando espaço nesse cenário. Porém, nem sempre os computadores estão prontamente disponíveis em todas as escolas. É por isso que pesquisadores vêm realizando estudos empíricos sob a perspectiva do desenvolvimento de princípios de PC ou da computação por meio de atividades desplugadas.

Atividades simples, como desenhar, resolver problemas ou enigmas, cortar, fazer uso de cartões, fazem parte da computação desplugada, que contribui para a aprendizagem de conceitos de ciência da computação. Ressaltamos que elas são direcionadas constantemente para ensino e aprendizagem mais cenestésicos, de aprender fazendo, colocando a mão na massa.

Em Brackmann *et al.* (2017), é realizada uma pesquisa aprofundada e baseada inteiramente em atividades desplugadas aplicadas para crianças na Educação Básica do Brasil e da Espanha. Em sua essência, foram obtidos resultados positivos em relação ao uso desse tipo de atividade para o desenvolvimento de habilidades do PC. Foram realizados testes com crianças de 5º e 6º anos do Ensino Fundamental, para os quais houve melhora significativa nas atividades escolares, quando fizeram uso de PC desplugado.

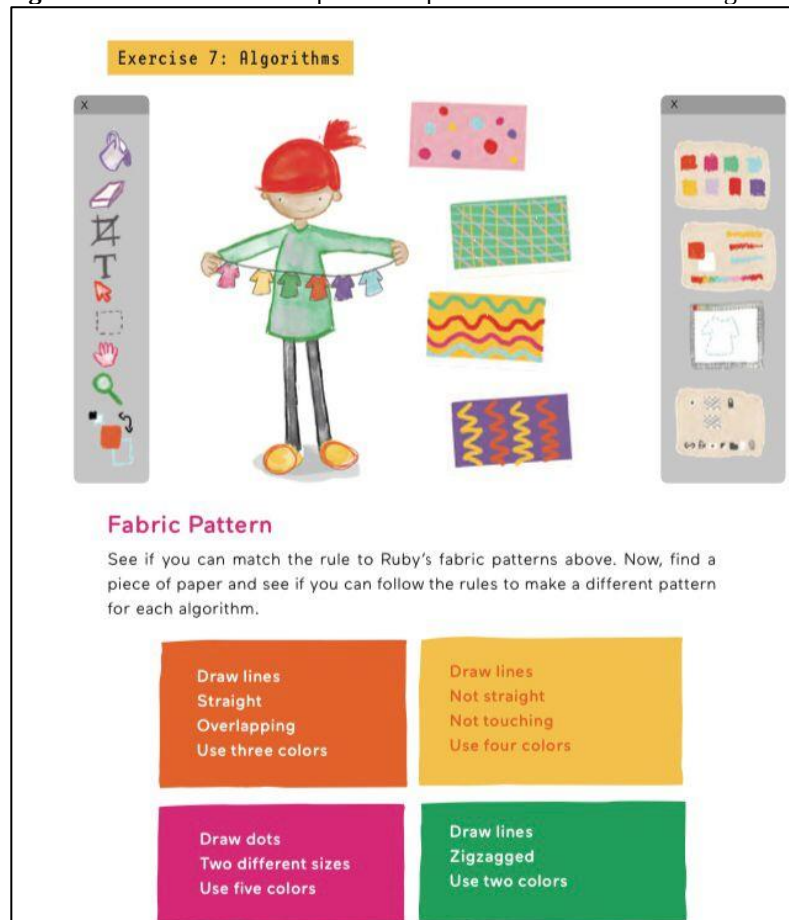
De acordo com Vieira, Passos e Barreto (2013), existem algumas técnicas ou estratégias para a aplicação da computação desplugada.

- Não utilizar computadores.
- Ensinar ciência da computação.
- Aprender fazendo.
- Garantir que o ensino seja divertido.
- Não utilizar equipamento.
- Desenvolver aplicação das técnicas.
- Permitir que qualquer pessoa faça uso.
- Reforçar a cooperação, a comunicação e a solução de problemas.
- Ser flexível a erros, pois pequenos erros não podem impedir a compreensão.

Há diferentes maneiras de trabalhar atividades desplugadas. Um exemplo é *Hello Ruby: Adventures in Coding* (tradução livre: *Olá, Ruby - Uma Aventura pela Programação*), de Liukas, (2015), um livro infantil ilustrado em que as crianças são apresentadas, por meio de histórias, a conceitos básicos de computação e codificação, sem a necessidade de falar de linguagem de programação. Ao longo da leitura, são propostas muitas atividades de “codificação” baseadas em jogos e atividades do dia a dia da criança. O livro constrói algumas analogias de fácil compreensão e quebra o paradigma de que programar é um ato difícil.

Em uma das atividades proposta por Liukas (2015), o leitor deve ligar os conjuntos de regras apresentados na Figura 24 a uma das estampas da personagem Rubbi, conseguindo assim criar uma estampa para cada algoritmo. Nesse tipo de atividade, podemos observar conceitos de computação e de matemática por meio de RP. À medida que se criam estampas e são aplicadas as regras, pode-se observar o emprego dos conceitos de reta, combinação, polígono, não polígono, permutação e arranjos, entre outros.

Figura 24 - Montando estampas de roupas - desenvolvimento de algoritmo



Fonte: Liukas (2015, p. 79).

Independentemente de conhecerem qualquer linguagem de programação, os estudantes devem desenvolver a capacidade de resolver problemas; quanto mais problemas resolverem, mais bem estruturados serão seus algoritmos. De fato, o pensamento algorítmico pode ser introduzido com sucesso sem a ajuda de um computador (BELL *et al.*, 2005).

Os referidos autores realizaram um estudo relacionado à popularização da matemática na ciência da computação desplugada, disponibilizando em CS Unplugged, que engloba várias atividades de ensino e aprendizagem de ciência da computação por meio de jogos e quebra-cabeças. Tais atividades podem viabilizar o ensino em sala de aula, fazendo com que o aluno pense em questões complexas sem ter que programar.

Na atividade demonstrada a seguir, extraída de Bell *et al.* (2005), consta com a imagem ampliada da letra A representada por *pixels*, em que cada ponto consiste em um *pixel*, que pode ser representado por números. A partir do momento que o aluno compreende os códigos, são propostos a ele quadros para decodificação de diferentes figuras. Assim mostra a Figura 25, indo de códigos de níveis básico a avançados.

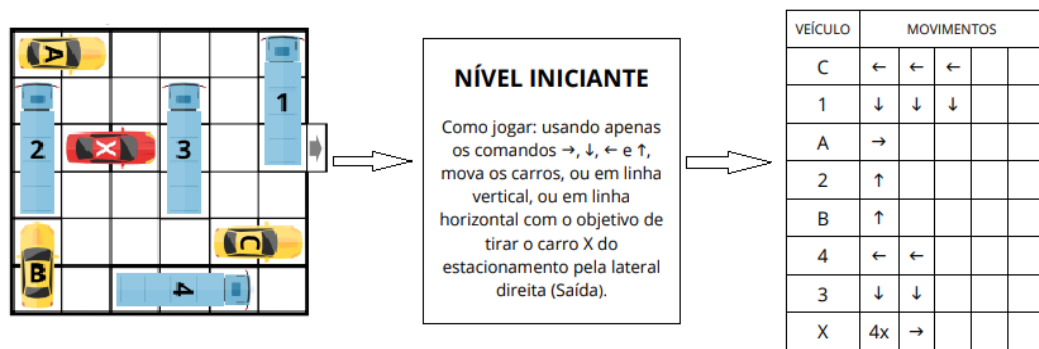
Figura 25 - Exemplo de códigos

	■	■	■		1, 3, 1
				■	4, 1
	■	■	■	■	1, 4
■				■	0, 1, 3, 1
■				■	0, 1, 3, 1
	■	■	■	■	1, 4

Fonte: Bell *et al.* (2005).

O site Computacional.com.br, uma iniciativa do pesquisador e mantenedor Brackmann (2017) com base em sua tese de doutorado, traz diferentes informações acerca de PC e disponibiliza atividades desplugadas que podem ser desenvolvidas em sala de aula. Entre as atividades disponíveis, pode-se citar a do Estacionamento Algorítmico, a qual auxilia o aluno na organização de informações e compreensão delas por meio de comandos fornecidos para que atinja as cartas-desafios.

Na Figura 26, há uma demonstração de uma carta-desafio constituinte da atividade desplugada proposta para o aluno, com as regras para desenvolvimento da atividade e o código final para a carta. Esse é um exemplo de como funciona o jogo, lembrando que o grau de dificuldade dos desafios vai aumentando conforme o aluno vai avançando.

Figura 26 - Jogo do estacionamento algorítmico – desafio 1

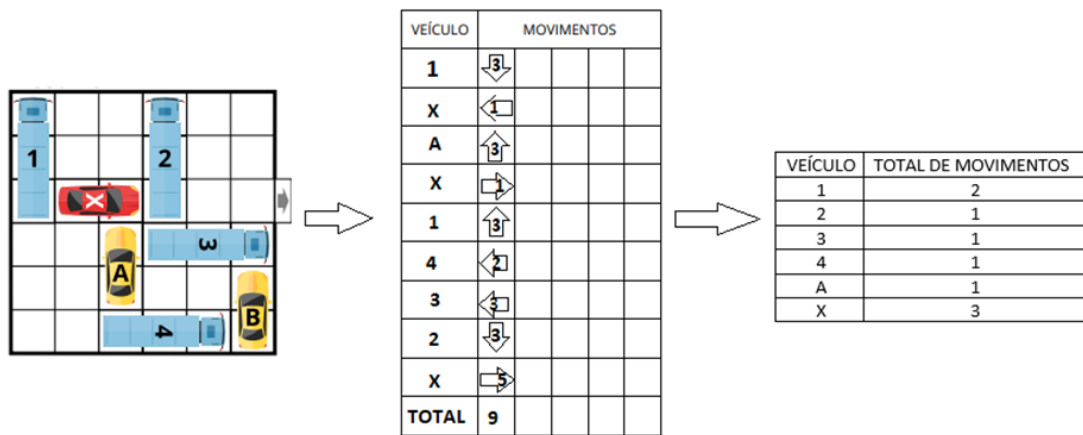
Fonte: adaptada de plataforma Computacional¹³.

Na Figura 26, é apresentada a carta-desafio 4 do jogo Estacionamento Algorítmico, do site Computação.com.br, em que se propõe uma nova maneira de realizar a atividade com alunos da Educação Básica. A forma exposta a seguir busca fazer com que o aluno desenvolva

¹³ O link de acesso para a plataforma é www.computacional.com.br.

habilidades matemáticas — algumas explícitas, como quando fazemos uso do mecanismo do jogo, e outras implícitas, como o desenvolvimento e a utilização de estratégias de raciocínio. Desse modo, o aluno começa a formular perguntas pertinentes para compreender melhor seu problema, determinando objetivos secundários para alcançar o resultado almejado sem agir de forma impulsiva.

Figura 27 - Jogo do estacionamento algorítmico – D



Fonte: adaptada de plataforma Computacional.

No caso da Figura 27, foram propostas questões antes do desafio, oportunizando o desenvolvimento do método de tentativa e erro ao se sugerir que o aluno aproximasse a situação da vida real. Vejamos algumas dessas questões:

- O que impede o carro x de se movimentar?
- O que impede o carro A?
- É possível movimentar o carro 1 para liberar o vermelho?

Outras perguntas poderiam ter sido realizadas para que, ao longo do processo, o aluno reduzisse seus movimentos e melhorasse sua estratégia ao pensar no problema quebrando-o em pedaços. Liukas (2015) caracteriza essa etapa como decomposição.

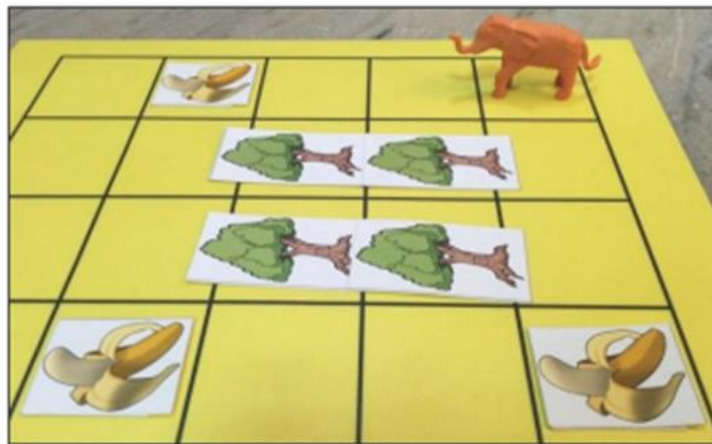
Algumas maneiras interessantes de introduzir as habilidades do PC com atividades desplugadas foram exploradas na Índia por meio de atividades *maker*¹⁴ (Faça você mesmo) de construção de ferramentas de treinamento usando materiais de baixo custo e de *kits* de circuitos suaves que podiam ser vendidos como *kits* educacionais em escolas (UNNIKRISHNAN, 2016).

Por se tratar de uma comunidade com escassez de recursos na área rural, Unnikrishnan (2016) propõe jogos com computação desplugada, nos quais o personagem principal da narrativa é um elefante, animal comum naquela região. O jogo se chama *Haathi Mera Saathi* (tradução

¹⁴ *Maker*: fazer com as próprias mãos; aprendizagem com construção.

livre: *Meu Amigo Elefante*) (Figura 28). Ele foi projetado para reproduzir a linguagem de programação *Processing*, também utilizada na metodologia desta tese. O jogo acontece em grades, em que são posicionados diferentes obstáculos, como árvores e bananas. Com o conjunto de comandos em mãos, o jogador cria passos — isto é, algoritmos — que permitirão que o elefante possa se mover e comer bananas. Conforme o aluno vai conseguindo resolver os desafios, a quantidade de obstáculos é ampliada, aumentando o grau de dificuldade do jogo.

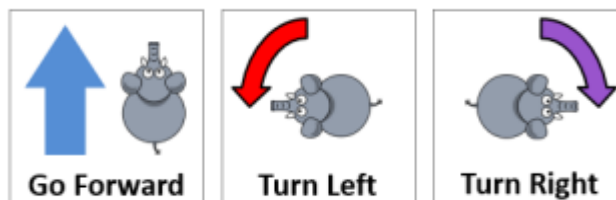
Figura 28 - Modelo do tabuleiro do jogo com obstáculos (Haathi Mera Saathi)



Fonte: Unnikrishnan (2016).

O jogo da Figura 29 inicia-se com um mestre, que compõe o tabuleiro com bananas e árvores. Em seguida, o jogador cria um conjunto de comandos (Figura 30), que direcionará o elefante a comer as frutas. O jogador só mostrará o cartão-resposta quando tiver solucionado a questão, lembrando que o elefante deve comer todas as bananas. Caso seja necessário, o jogador pode realizar várias tentativas para resolução, criando assim vários algoritmos.

Figura 29 - Exemplos de cartões de comando do jogo (Haathi Mera Saathi)



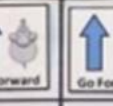
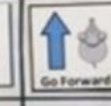
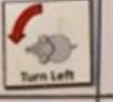
Fonte: Unnikrishnan (2016).

Na Figura 30, podemos visualizar todo programa realizado pelo jogador para resolver seu problema, em que o aluno faz uso de diferentes conceitos computacionais, como *loops*¹⁵ (quando

¹⁵ *Loops*: execução sequencial que é interrompida a partir de uma condição.

o elefante realiza o movimento várias vezes), condicionais, comandos e variáveis ao criar sua tabela de programação.

Figura 30 - Tabela de programação

 Go Forward	 Go Forward	 Eat	 Turn Left	 Go Forward	 Go Forward	 Eat	 Turn Left	 Go Forward	 Go Forward
 Eat	 Turn Left								

Fonte: Unnikrishnan (2016).

Na tabela de programação e em todo o desenvolvimento do jogo, é possível observar características do PC, por exemplo, quando o jogador divide suas fichas para iniciar os movimentos ou ainda quando cria seu programa para resolver o problema. Pode-se afirmar o mesmo em relação aos conceitos matemáticos, que surgem desde o início das funções propostas, como na confecção do tabuleiro por meio de coordenadas, giros, pensamento lógico e estratégico para pensar a utilização de comandos e a escrita do algoritmo final. Cria-se, assim, um grande plano de fundo para a introdução do *Processing*.

CAPÍTULO 3 – PROCEDIMENTOS METODOLÓGICOS DA PESQUISA

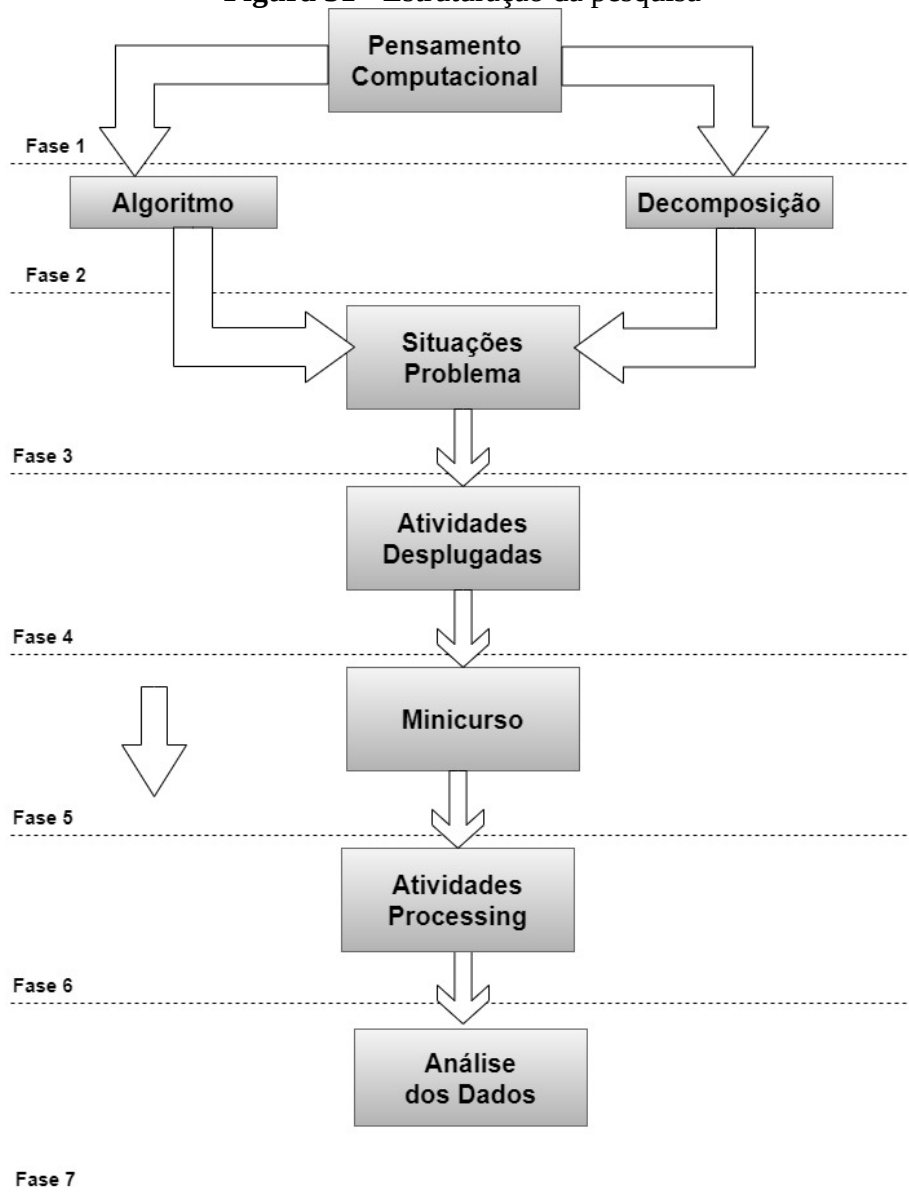
Este capítulo aborda a organização metodológica da pesquisa. A análise de dados e todos os procedimentos metodológicos têm como objetivo responder ao seguinte conjunto de questionamentos, já abordados em nossa discussão da problemática da pesquisa:

- Como acontece o processo de RP realizado pelo estudante?
- O que podemos observar nesse processo?
- Qual seria a melhor maneira de resolver um problema no âmbito do ensino da matemática, independentemente da sua complexidade?
- Conseguimos perceber na RP habilidades do PC?

3.1 Delineamento da pesquisa

O presente trabalho foi dividido em quatro etapas, como demonstra a Figura 31. Primeiramente, foi realizada a revisão da literatura, sendo selecionadas publicações que conceituavam aspectos importantes de PC e RP relacionados ao ensino de matemática, que formam o referencial teórico desta tese. Também foram analisadas algumas atividades desplugadas e o ambiente de aprendizagem *Processing*.

A Figura 32 representa a estrutura de da pesquisa propriamente dita. Buscamos apresentar a forma como os dados foram coletados, facilitando assim a conferência das evidências e sua possível reprodutibilidade em futuras pesquisas que visem a aprofundar o objeto aqui estudado.

Figura 31 - Estruturação da pesquisa

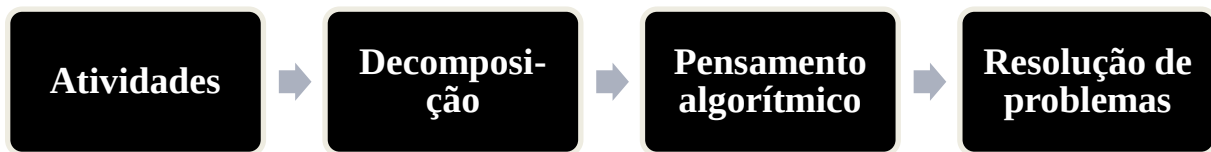
Fonte: desenvolvida pela autora (2022).

Outra forma de visualizar a pesquisa é por meio da consideração do conceito de variáveis, entre as quais destacamos:

- variável decomposição;
- variável pensamento algorítmico;
- variável resolução de problemas (RP).

A relação entre as variáveis nos leva ao esquema da Figura 32, em que o mecanismo para se chegar à RP passa primeiramente pela decomposição e, na sequência, pela junção de suas partes na resolução algorítmica.

Figura 32 - Relação entre as variáveis do estudo



Fonte: desenvolvida pela autora (2022).

3.2 Metodologia de trabalho

Os dados para este estudo vêm da observação do desempenho dos alunos em resolução de problemas matemáticos, utilizando como estratégia competências do PC, enfatizando as habilidades de decomposição e pensamento algorítmico.

A pesquisa é de cunho qualitativo. O método de pesquisa qualitativa representa um modo indutivo de análise da pesquisa ou um processo de mudança de observações específicas para uma teoria geral. Na pesquisa qualitativa, os dados são compreendidos de forma não-numérica, ou seja, por avaliação do sentido qualitativo ali mencionado. Geralmente nas pesquisas qualitativas a quantidade de amostras é menor que na pesquisa quantitativa, podendo o pesquisador estar mais próximo do participante da pesquisa (VOLPATO ET AL., 2013).

Dessa forma, a abordagem aqui apresentada apresenta aos leitores como obtemos os dados, o que garante dois aspectos importantes: evidências interindividuais e perenes. As primeiras garantem que outros pesquisadores podem usufruir dos procedimentos adotados e tecer conclusões que ampliem o olhar das informações apresentadas neste trabalho. O segundo aspecto, a perenidade das evidências, é destacável na medida em que exploramos dados que balizam conclusões válidas para a ciência. Nesse contexto, lembramo-nos sempre do método científico como mantenedor de conhecimento provisório, ou seja, a melhor explicação válida até o momento, podendo e devendo, assim, ser acrescida e modificada no futuro.

3.3 Participantes e local da pesquisa

A pesquisa foi desenvolvida com estudantes dos anos finais do Ensino Fundamental, com idade média entre 13 e 15 anos, pertencentes a escola da rede pública de ensino regular, com funcionamento de três turnos, com 608 alunos matriculados, localizado em uma cidade de pequeno porte, no interior de São Paulo. Cujo currículo abarca seis aulas semanais de matemática e uma aula semanal de tecnologia, de 50 minutos.

Os participantes foram convidados pela pesquisadora a participar do estudo, no momento do convite foi explicado a necessidade da presença do aluno no período contrário que estudam, sendo essa a única exigência, devido ao minicurso e a realização das atividades laboratoriais.

Entre duas salas de 35 e 34 alunos, do 9º ano do Ensino Fundamental, nove alunos aceitaram fazer parte do estudo de imediato, porém um desistiu devido a uma oportunidade de um emprego. Percebe-se que muitos alunos da comunidade, auxiliam seus pais em casa com os cuidados de filhos menores e outros afazeres domésticos o que impossibilitou a participação em massa dos alunos convidados. É importante ressaltar que todos os cuidados éticos foram adotados e as identidades dos participantes foram preservadas.

Após verificado o aceite dos participantes e recebidas as autorizações dos responsáveis, que foram até a escola para assinarem o termo de consentimento e tomarem ciência da pesquisa, os alunos receberam um *link* para responder a um formulário, por meio do Google Forms, para que pudessemos conhecer algumas características do grupo voluntário.

Os estudantes foram classificados de acordo com suas respostas, conforme o Quadro 10. A partir desse ponto, teve início a primeira etapa das atividades.

Quadro 10 - Classificação dos participantes na primeira etapa das atividades

Alunos	Idade	Disciplina com maior afinidade	Sexo	Quantidade de horas diárias de uso de recursos tecnológicos	Conhecimento em inglês
Aluno1	15	Língua Portuguesa	M	10h	Intermediário
Aluno 2	14	Matemática	F	10h	Básico
Aluno 3	15	Matemática	M	8h	Intermediário
Aluno 4	14	Inglês	F	8h	Avançado
Aluno 5	14	Língua Portuguesa	F	6h	Intermediário
Aluno 6	15	Matemática	F	8h	Básico
Aluno 7	14	Matemática	M	9h	Intermediário
Aluno 8	15	Inglês	F	11h	Avançado

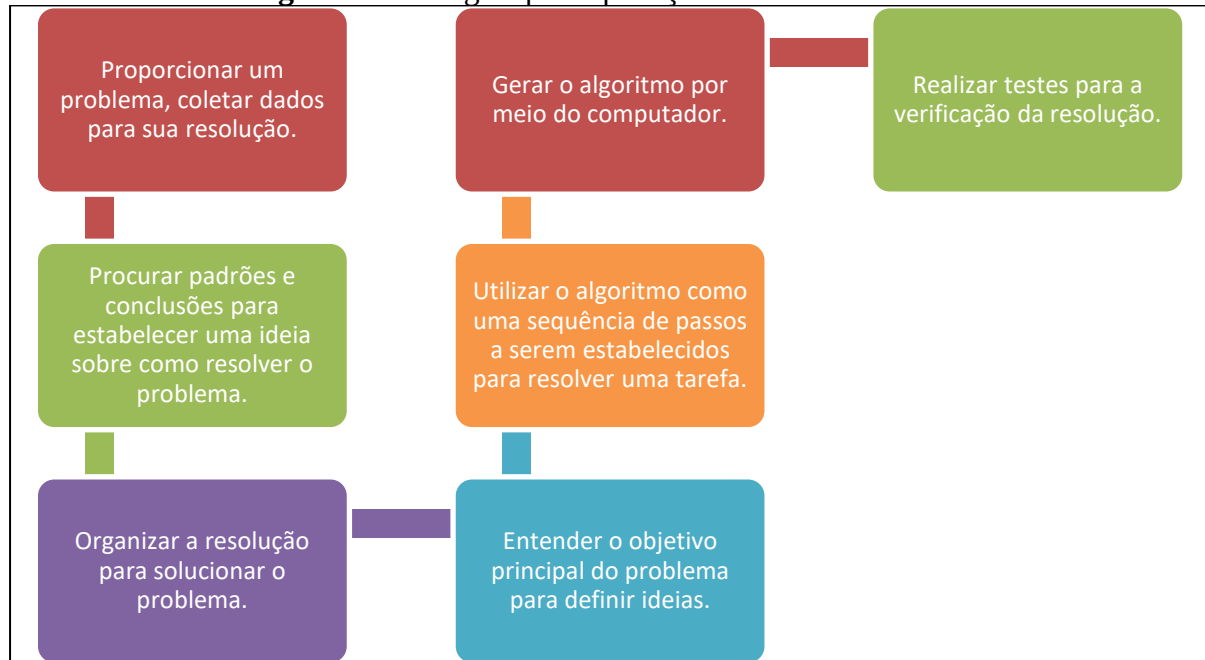
Fonte: desenvolvida pela autora (2022).

As horas de uso diárias de recursos tecnológicos estão relacionadas a horário de estudo diário, leitura e entretenimento. Constatamos foco maior em redes sociais, séries, jogos, mangás e animes.

3.4 Organização da coleta de dados

Existem estágios importantes para aplicação do PC na educação matemática (Figura 33). São eles:

Figura 33 - Estágios para aplicação do PC em matemática



Fonte: desenvolvida pela autora (2022).

Tais estágios nos permitem perceber que tanto o PC como o pensamento matemático são formas de engajamento no âmbito de RP e podem e devem ser aprimorados pela prática por meio de discussões e reflexões. Uma vez que esses estágios são compreendidos e colocados em prática, são desenvolvidas estratégias para modelar a RP de forma eficiente, de modo que podemos nos deparar com resoluções mais abrangentes para diferentes classes de problemas.

Foi pensando nesses estágios que as atividades do minicurso foram estruturadas. Os itens que configura Quadro 11 sintetizam, de maneira prática, como ele foi direcionado. Por meio de nove aulas de uma hora no laboratório de informática da instituição escolar, entre os dias 14 e 21 de fevereiro de 2020, foram explicados aos estudantes fundamentos do PC e sua abordagem para RP com cálculos utilizando algoritmos.

O minicurso foi ministrado pela autora. Nas aulas desenvolvidas no *Processing*, contamos com a participação do professor Dr. Edriano Carlos Campana, que auxiliou com a parte de supervisão em problemas computacionais relacionados a códigos. Isso foi necessário pois, dessa forma, ganhamos tempo, e os alunos puderam tirar dúvidas pertinentes em momento síncrono.

Quadro 11- Estrutura do minicurso

MINICURSO		
	Conteúdo	Objetivo
Aula1	Teoria: habilidades do Pensamento Computacional	Compreender o conceito e adquirir competências básicas do PC.
Aula 2	Resolução de problemas lógicos: verificando conhecimento matemático	Verificar o conhecimento lógico do aluno e encontrar habilidades do PC na resolução de situações matemáticas.
Aula 3	Atividades de decomposição e pensamento algoritmo (mão na massa)	Resolver atividade desplugada, fazendo uso de habilidades do PC.
Aula4	Introdução ao <i>Processing</i>	Explicar, com exemplos práticos, os objetivos do ambiente de programação.
Aula5	Conceito de variáveis em programação e seus tipos	Resolver algoritmos com uso de tipos diversos de variáveis.
Aula 6	Conceitos de entrada, processamento e saída de dados	Aplicar algoritmos que emitam as mensagens de entrada de dados, em seguida o processamento e exibição dos resultados.
Aula 7	Operadores aritméticos, relacionais e lógicos	Compreender a aplicação dos diferentes tipos de operadores na resolução de algoritmos.
Aula 8	Estruturas condicionais	Resolver algoritmos com as estruturas <i>if</i> e <i>case</i> , utilizadas quando o resultado do algoritmo pode implicar várias possibilidades.
Aula9	Estruturas de repetição	Utilizar os comandos <i>for</i> e <i>while</i> na resolução de algoritmos com cálculos que contenham repetição, como tabuada ou fatorial, por exemplo.

Fonte: desenvolvida pela autora (2021).

3.5 Desenvolvimento do minicurso

As atividades do minicurso, descritas no Quadro 11, buscam preparar os alunos para a realização das atividades a serem analisadas.

- Aula1: foram abordados o conceito de PC, suas habilidades e aplicações. Por meio da exibição de um vídeo, foi estabelecido um diálogo entre participantes e feita a análise de atividades que demonstram as quatro habilidades do PC. Em seguida, os alunos responderam a um questionário inicial sobre PC.
- Aula 2: nesse dia as atividades envolveram RP, com o desenvolvimento do raciocínio lógico. As dinâmicas foram realizadas por meio da ferramenta do Code.org. Os alunos puderam perceber o conceito de PC em situações-problemas.
- Aula 3: o objetivo dessa aula foi desenvolver os conceitos explanados anteriormente por meio de uma atividade desplugada. No caso, a atividade proposta consistiu na resolução do problema Torre de Hanói. Os alunos o resolveram sozinhos, cada um com o seu material concreto, produzido pela autora (como consta na Figura 35).
- Aula 4: focamos a apresentação do ambiente de programação do *Processing*, demonstrando, de maneira bem didática, como funciona seu IDE. Desenvolvemos um algoritmo bem simples, para que os alunos visualisassem, de forma mais prática, seu potencial futuro.
- Aula 5: partindo do pressuposto de que um importante componente para resolução de algoritmos em linguagens ou ambientes de programação é ter bem consolidado o conceito de variáveis, nesse momento criamos algoritmos simples para introduzir esse conceito.
- Aula 6: nas atividades dessa aula foram desenvolvidos algoritmos que demonstrassem, de maneira clara, que o computador em si é uma máquina que o tempo todo está processando informações e que os algoritmos são uma forma de fazer isso.
- Aula 7: para a resolução de algoritmos, é importante que o participante use operadores aritméticos, relacionais e lógicos. Dessa forma, trabalhamos nessa aula algoritmos que explorassem o uso desses operadores em situações específicas. Embora os operadores aritméticos sejam mais utilizados, os relacionais e lógicos desempenham importante função no pensamento algorítmico.
- Aula 8: um ponto muito importante, em qualquer aprendizado de programação, é entender o conceito de estruturas condicionais, conhecidos amplamente pelo comando *if* (“se”, em português). Sem notar, ao utilizarmos aplicativos, estamos o tempo todo tomando decisões ao clicar em determinadas opções. Sendo assim, essa aula explorou algoritmos

que podiam exibir uma resposta dentre duas ou mais para o usuário, dependendo das informações processadas.

- Aula 9: na última aula, foram apresentadas as estruturas de repetição (*for* e *while* – “para” e “enquanto”, em português). Elas são muito relevantes, pois possibilitam a repetição de cálculos e operadores lógicas com poucas linhas de programação. Sem o implemento de tais estruturas, alguns algoritmos poderiam ter milhares de linhas, por isso entendemos que era vital ao participante compreender com exemplos como se aplicam as estruturas de repetição em algoritmos.

As aulas foram desenvolvidas conforme o cronograma explicitado no Quadro 12. O minicurso foi composto por aulas expositivas e práticas, de forma que os estudantes podiam praticar habilidades do PC dentro dos temas abordados.

Quadro 12 - Cronograma do minicurso

Etapas	Fevereiro/2020					
	Dia 14	Dia 17	Dia 18	Dia 19	Dia 20	Dia 21
Aula 1	X					
Aula 2		X				
Aula 3		X				
Aula 4			X			
Aula 5			X			
Aula 6				X		
Aula 7					X	
Aula 8						X
Aula 9						X

Fonte: desenvolvido pela autora (2020).

Em seguida, durante o mesmo minicurso (Quadro 13), focamos exercícios de resolução de algoritmos utilizando a linguagem de programação *Processing* (sem licença de uso), que podia tanto ser utilizado em versão *desktop* (instalado diretamente no próprio computador) ou na versão Web (em que o aplicativo é aberto via navegador, como o Google Chrome, em ambiente de computação em “nuvem”).

A princípio, os experimentos foram conduzidos de forma desplugada, ou seja, sem a necessidade de dispositivos eletrônicos, como *smartphones*, *tablets* ou *laptops*. Em seguida, a linguagem de programação citada no parágrafo anterior foi introduzida por meio de exercícios que envolviam primeiramente a realização de cálculos matemáticos (Figura 32).

Figura 34 - Aplicação do minicurso

Fonte: desenvolvida pela autora (2022).

O foco das atividades foram a decomposição e o pensamento algorítmico, selecionados visando à coleta de dados para melhor entendimento desses conceitos pelos estudantes.

Ministradas as nove aulas propostas (figura 34), propusemos uma mesa redonda, em que os participantes puderam tirar dúvidas e fazer questionamentos. Dois dias depois, iniciamos as atividades (Quadro 13) com o uso do ambiente de aprendizagem *Processing*, que foram utilizadas para a análise dos dados desta pesquisa,

Quadro 13 - Atividades desenvolvidas no Processing

Atividades	Habilidade
1- Calculadora simples	Pensamento algoritmo, decomposição e reconhecimento de padrões e abstração
2- Verificação de mês do ano pelo número	Pensamento algoritmo, decomposição e reconhecimento de padrões e abstração
3- Soma de números de 0 a 10	Pensamento algoritmo, decomposição e reconhecimento de padrões e abstração
4- Tabuada	Pensamento algoritmo, decomposição e reconhecimento de padrões e abstração

Fonte: desenvolvida pela autora (2021).

CAPÍTULO 4 – ANÁLISE E DISCUSSÃO

Neste capítulo apresentamos os dados coletados na pesquisa, por meio de uma discussão que responde à pergunta desta tese.

4.1 Perspectiva geral

A aplicação relacionada ao desenvolvimento da presente pesquisa ocorreu no 9º ano do Ensino Fundamental de uma Escola Estadual do interior de São Paulo, tendo por objetivo analisar como os estudantes realizam a RP: se fazem uso de habilidades do PC como estratégia ou mecanismo na hora da resolução, ou ainda se conseguem contextualizar conhecimentos já adquiridos na hora de resolver um problema. A partir disso, verificamos a possibilidade de incluir o PC em suas resoluções e verificar como a sua utilização poderia modificar a maneira de resolver problemas, ao mesmo tempo que poderíamos fazer com que o estudante, após conhecer as habilidades do PC, fizesse uso dessas habilidades ou mecanismos do PC, trabalhando sua reutilização em diferentes problemas, para assim conhecer as quatro habilidades e usá-las na hora de resolver qualquer problema, independentemente do seu grau de complexidade. O intuito era melhorar sua prática de RP, uma vez que, como os índices das avaliações externas mostram, tais habilidades estão em defasagem na Educação Básica brasileira.

Os instrumentos utilizados para a coleta dos dados, foram questionários, análise de vídeos das atividades realizadas, diário de bordo, onde foi registrado de maneira precisa todo o desenvolvimento das atividades laboratoriais desenvolvidas pelos participantes.

Na etapa 1, antes do início do minicurso, foi enviado aos alunos um *link* para que respondessem a um questionário, de modo que pudéssemos conhecer um pouco dos sujeitos da pesquisa (Quadro 10) e suas aptidões relativas ao processo de ensino-aprendizagem em matemática.

Na etapa 2 do minicurso, foram ministradas três aulas de 50 minutos sobre as habilidades do PC, seus aspectos no ensino de matemática e recorrências ao se resolver problemas. Nesse momento os alunos puderam resolver algumas situações-problemas de nível básico de forma desplugada, para compreender as quatro habilidades do PC. A atividade desplugada desenvolvida pelos alunos foi a Torre de Hanói, cujo tabuleiro (como consta na Figura 35) lhes foi ofertado. Os dados dessa atividade farão parte das análises posteriores.

As habilidades foram trabalhadas considerando as perspectivas dos autores citados no referencial teórico desta pesquisa. Uma delas foi a abstração, que, para Wing (2006), é uma das mais importantes, pois é utilizada de diferentes maneiras ao se resolver um problema, desde a

escrita de um algoritmo até a organização da estrutura em sistemas. Outra habilidade explorada foi o reconhecimento de padrões, que, segundo Liukas (2015), consiste em encontrar padrões similares para se resolver problemas simples ou complexos de forma eficiente. Também analisamos a decomposição, que envolve a fragmentação de um problema em partes menores (LIUKAS, 2015), e, por último, o algoritmo, que pode ser entendido como um conjunto de instruções claras e objetivas para se resolver um problema.

Ainda na etapa 2, como consta no Quadro 11, a partir da aula 4 do minicurso, apresentamos o ambiente de aprendizagem *Processing* e outros assuntos relevantes para pesquisa. Nessa fase as oficinas tiveram duração de seis aulas de 50 minutos e foram destinadas apenas aos oito participantes da pesquisa. Nesse momento os participantes aprenderam conceitos de programação.

Na etapa 3, após a finalização do minicurso, os alunos realizaram quatro atividades, de maneira conectada, no ambiente de aprendizagem *Processing*, por meio da versão *desktop* do aplicativo. O foco de tais atividades foram as habilidades de decomposição e pensamento algoritmo. Ressaltamos aqui que, embora as outras habilidades do PC também estivessem presentes — afinal todas se interligam em vários momentos —, não eram elas a fonte de análise para esta pesquisa.

No referencial teórico deste trabalho, podemos observar, nas Figuras 16, 17, 18 e 19, exemplos de aplicações que podem ser desenvolvidas no ambiente *Processing*. Essas atividades requerem um pouco de conhecimento sobre o manuseio da ferramenta e alguns conceitos fundamentais de programação, que os alunos puderam desenvolver durante o minicurso.

Na data determinada pelo cronograma da Quadro 12, as atividades tiveram início. Explicamos aos participantes a importância de as oficinas serem gravadas (vídeo e áudio), para futura análise e elaboração dos resultados desta tese.

As atividades do Quadro 13 foram realizadas durante três dias consecutivos, com blocos de aula com duração de 2 horas 40 minutos. Serão elas as fontes do resultado desta tese.

4.1.1 Etapas desenvolvidas

Na etapa 1, os alunos responderam a um questionário que, no tópico 4.2, nos possibilitou analisar as seguintes características dos participantes: idade, disciplina preferida, uso de tempo na internet e habilidades com a língua inglesa. Nesse mesmo questionário, uma questão interessante para ser analisada diz respeito ao gosto do estudante pela matemática, aos motivos de suas preferências e às suas facilidades ou dificuldades. No Quadro 14, estão elencadas as respostas dos participantes.

Quadro 14 - Gosta de matemática/motivo/dificuldades ou facilidades

Participante 1	“Não gosto de matemática, pois tenho mt dificuldade de aprender essa matéria.”
Participante 2	“Sim, porque eu gosto de fazer cálculos, muitas vezes tenho facilidades, acho que é uma matéria sem muita frescura, exata.”
Participante 3	“Sim, sempre gostei quando pequena gostava porque me desafiava, hoje porque tenho facilidades.”
Participante 4	“Não sou muito fã, nada em especial, mas tenho facilidade em resolver seus problemas.”
Participante 5	“Não muito, acho abstrata demais, o tempo vai passando e vou tendo mais dificuldade.”
Participante 6	“Sim, mais as vezes eu tenho alguns problema para resolver algumas atividades, mais não desanimo, pois quero aprender e diminuir ad dificuldades.”
Participante 7	“Eu gosto, pra mim não tem mistério é pura lógica, sempre tive facilidades.”
Participante 8	“Um pouco, se não tivesse que memorizar tantas fórmulas seria mais dá hora, até que aprendo fácil.”

Fonte: desenvolvida pela autora (2022).

No contato inicial com os alunos frente a um assunto de extrema relevância para pesquisa, — já que o proposto aqui é trabalhar conceitos matemáticos —, percebe-se que aqueles que têm mais facilidade com a matemática a elegem como seu componente curricular favorito. Chegamos a essa conclusão ao compararmos o Quadro 10, que classifica os estudantes, com o Quadro 14. Os alunos 2, 3, 6 e 7 relatam achar a disciplina fácil e a indicam como uma de suas favoritas.

De maneira geral, os estudantes que dizem não gostar da disciplina justificam esse fato pela forma com que ela é trabalhada: por exemplo, o participante 5 diz que “as vezes fazemos dez contas iguais, isso é chato”. Recuperando Fiorentini (1995), o professor ensina matemática por memorização e repetição, causando exaustão no aluno. Por isso, a disciplina deveria ser desenvolvida por meio de RP, de forma reflexiva.

Uma resposta relevante a ser analisado foi a do participante 4, que afirma ter “facilidade em resolver problemas”. De acordo com ele, quando é contada uma história por trás da

matemática, fica mais fácil de entendê-la, “talvez por nem sempre ter isso, acaba não me atraindo tanto”. Quando o aluno menciona história, dá a entender a necessidade de contextualização do conteúdo.

Nota-se, na resposta do participante 5 — “abstrata demais, o tempo vai passando e vou tendo mais dificuldade” —, que, conforme os anos vão avançando, a matéria vai ficando menos lúdica, e a partir disso surgem novas dificuldades. Quando questionado o que é abstrato para ele, o participante relata “tipo assim, fórmula de Bhaskara”. Ele se refere à reprodução da fórmula e ao fato de não saber o que aquilo significa e onde será utilizado.

Por fim, de forma geral, nota-se que os alunos, mesmo os que afirmam ter um pouco de dificuldade com a matemática, têm um olhar otimista em relação às atividades abordadas na pesquisa.

Após a realização das aulas de 1 a 3, os participantes responderam ao questionário 2, sobre a importância e relevância do PC na RP. Seguem as respostas no Quadro 15.

Quadro 15 - Integração de habilidades do PC na RP

Participante 1	“Realmente é uma boa estratégia, a abstração foi a mais complexa a meu ver.”
Participante 2	“Saber que posso utilizar a mesma estratégia na hora de resolver um problema é muito bom.”
Participante 3	“Eu tento entender um problema, antes de resolver. Mais nunca pensei em separar em partes, vai me ajudar muito.”
Participante 4	“Nunca pensei em resolver problema utilizando estratégia, só ia lá e resolvia. A decomposição já fazia sem saber.”
Participante 5	“Com certeza vai mudar o meu modo de resolver um problema, vou fazer uso das quatro habilidades do Pensamento Computacional.”
Participante 6	“Estou na expectativa para utilizar o Pensamento Computacional no dia a dia, acho que vai me ajudar, foi bom conhecer.”
Participante 7	“Nossa, massa isso. É ver o problema de um jeito diferente. Acho que vai me ajudar nas outras disciplinas. Saber que na maioria das coisas existe um padrão é uma visão diferente para mim.”
Participante 8	“Entendi o que o pensamento computacional ajuda na matemática, espero poder utilizar a partir de agora na hora que eu ver um problema.

	O algoritmo construído vai me ajudar a ver o problema inteiro, talvez diminua minha dificuldade.”
--	---

Fonte: desenvolvido pela autora (2022).

Ao longo das atividades que abordavam os conceitos e as habilidades de PC, os alunos foram questionados a respeito de diferentes situações, para que pudéssemos observar se houve uma compreensão mínima dos conceitos de PC e dos pilares utilizados como fonte da pesquisa. É certo que alguns tiveram mais facilidades que os outros, ao se expressar em relação aos conceitos vistos. A real compreensão deles somente poderá ser constatada no decorrer das atividades. É possível fazermos algumas observações ao analisarmos algumas falas dos participantes, porém nada que certifique o seu grau de abstração do que foi demonstrado até o momento. Por exemplo, nota-se que, quando o participante 1 diz que abstração foi a mais difícil de entender, não é possível saber se ele entendeu o conceito de total ou parcialmente, por isso destacamos que as dúvidas serão analisadas no decorrer desta tese.

Durante as atividades, houve um momento em que as quatro habilidades do PC foram analisadas separadamente, e os alunos passaram a tentar identificá-las. Percebe-se que, a princípio, o reconhecimento de padrão e o algoritmo foram as mais apontadas, pois podem ser construídas e visualizadas. As seguintes falas validam essa conclusão: “gostei de pensar no algoritmo como uma receita” (participante 6) e “vamos criar vários passos a passo, assim podemos ver como tudo foi resolvido” (participante 2). Por meio dessas respostas, podemos constatar a compreensão do conceito de pensamento algorítmico: quando mencionam “receita” ou “passo a passo”, os participantes estão se referindo à construção do algoritmo, que, segundo Cormen *et al.* (2009), consiste em uma forma simplificada de resolução, por etapas, que utiliza a definição como uma sequência de passos computacionais que transforma dados de entrada em saídas.

Nota-se ainda, na fala do participante 3, a referência à decomposição como uma prática já utilizada por eles, embora nunca a tivessem reconhecido no ato da resolução. Ao dizer “separar em parte”, percebe-se a assimilação do conceito, já que a decomposição se caracteriza pela fragmentação dos problemas em partes menores e mais gerenciáveis.

Podemos dizer, então, que os alunos, de forma geral, ao conhecer os conceitos de PC e reconhecer suas habilidades dentro de RP, percebem que criar uma estratégia para resolução faz o problema ficar menos denso. Eles, além disso, notam a resolução não é feita de qualquer maneira, pois é gerada a partir de um contexto, desconhecido por eles até aquele momento. A resposta do participante 4 comprova essa conclusão: “Nunca pensei em resolver problema

utilizando estratégia, só ia lá e resolvia. A decomposição já fazia sem saber, já repartia o problema em partes menores, as vezes tirava dados que não iria utilizar”. Essa fala denota a habilidade de decompor os problemas em subproblemas, menores e mais gerenciáveis, facilitando assim a resolução.

4.2 Resultado da atividade desplugada

Na etapa 2, foi realizada a RP desplugada para que os participantes compreendessem, identificassem, desenvolvessem as habilidades dentro de situações-problemas. A atividade selecionada para isso foi a Torre de Hanói.

A Torre de Hanói (Figura 35) é um recurso de gamificação desplugada, transmitido no ano de 1883 pelo matemático francês Édouard Lucas. O objetivo principal desse recurso simples é realizar movimentos de discos de modo que a quantidade de movimentos seja a menor possível até atingir a última haste. Para isso, é necessário manusear as hastes centrais.

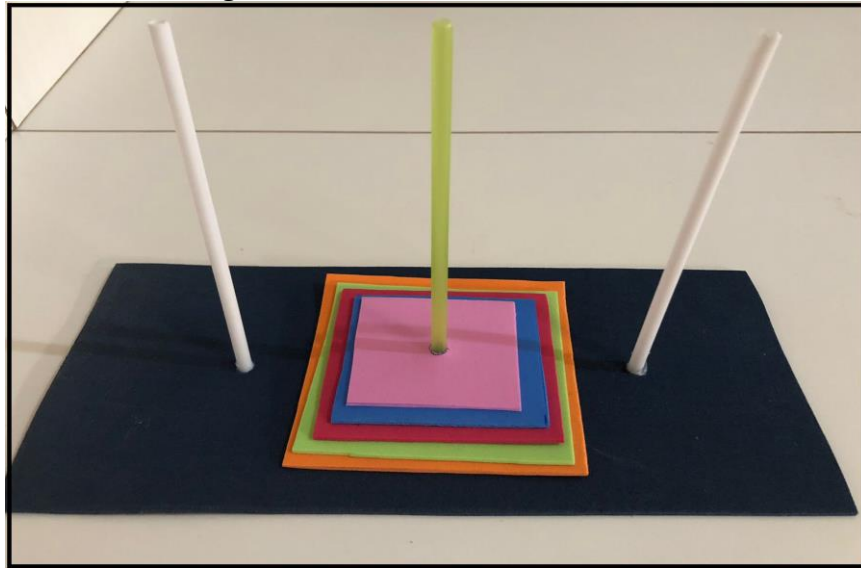
Quadro 16 - Atividade desplugada - torre de Hanói

Com quantos movimentos mínimos é possível resolver o problema da Torre de Hanói, independentemente da quantidade de discos?
Descrição
Transportar discos de uma haste a outra até atingir a haste final, com o menor número possível de movimentos. A torre enviada é composta de 3 hastes. Podem ser utilizados 3, 4, 5 discos, e assim por diante.
Habilidade do PC
Algoritmo: sequência de instruções para que o objetivo seja atingido. Decomposição: dividir o problema em unidades menores.

Fonte: desenvolvido pela autora (2020).

Ao fazer uso desse recurso como estratégia de aprendizagem, busca-se desenvolver no educando a habilidade de RP, assim como o raciocínio lógico matemático. Torna-se, assim, possível compreender seu processo de resolução e melhorar suas próprias estratégias ao se deparar com problemas com características semelhantes ou não.

Figura 35 - Tabuleiro - torre de Hanói

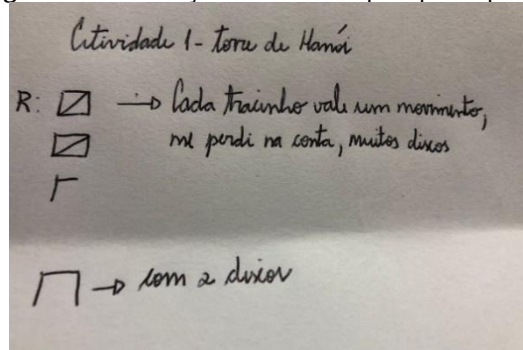


Fonte: Desenvolvida pela autora (2020).

Nesta atividade desplugada (Figura 35), referente ao desenvolvimento de habilidades do PC, as observações aqui darão ênfase maior à habilidade de decomposição e algoritmo, porém ressaltamos que a abstração e o reconhecimento de padrão também estão presentes na situação-problema.

No primeiro momento, os participantes tiveram contato com o jogo Torre de Hanói para se familiarizar com ele. Assim, foi permitido que manipulassem as peças nas hastes, compreendessem as regras e jogassem livremente. Nessa ocasião foi solicitado que o estudante transferisse o disco da haste A para a C, e assim por diante, fazendo o participante compreender todo o processo de funcionamento e buscar melhores estratégias para a resolução. Os participantes puderam manipular o jogo durante dez minutos e, depois desse tempo de reconhecimento e questionamentos, foi proposto que eles jogassem e realizassem a RP proposta no Quadro 16.

A partir de agora, vamos nos deter na resolução da atividade. Começaremos pelo participante 1.

Figura 36 - Resolução da atividade pelo participante 1

Fonte: resolução do participante 1; produção própria (2020).

Analisando a atividade anteriormente apresentada na figura 36, podemos observar que o participante 1 utiliza tracinhas para determinar cada movimento que realiza. Para ele isso melhora a visualização do seu registro. Nesse sentido, segundo o referencial teórico desta tese, podemos dizer que ele cria um algoritmo para resolver o problema.

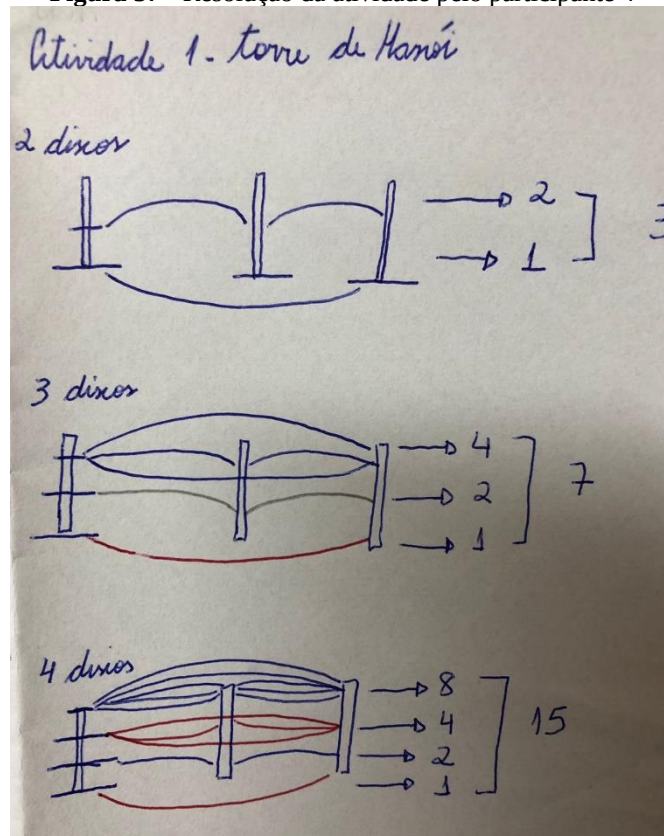
Quando questionado sobre sua forma de resolução, esse mesmo aluno relata que “cada traquinho é um movimento, no começo achei mais fácil”. Pode-se observar que os tracinhas são representados por um movimento até o quarto disco, a partir do qual ele faz uso da decomposição, de modo que cada grupo de dez tracinhas passa a corresponder a uma troca que se multiplica ao final, diminuindo assim o seu registro e a chance de erro. O participante afirma ter se perdido no registro, por isso passa a criar uma estratégia de pensamento lógico realizando a decomposição, cuja proposta é separar os problemas em partes menores. O aluno determina que uma dezena corresponde a cada dois grupinhos de tracinhas e, no final, multiplica por 6 e soma com os três tracinhas restantes. Ou seja, os seus problemas maiores foram separados em partes, além de registrados, o que permitiu que chegasse à resolução do problema.

Ao ser questionado, vemos que o aluno compreendeu o que foi solicitado, porém faltava estratégia na hora de realizar a resolução. Percebe-se que ele faz uso de tracinhas para transcrever os movimentos, porém acaba se perdendo nos próprios registros. Devido a isso, não consegue finalizar corretamente toda a situação-problema proposta. Contudo, quando a quantidade de discos é mínima, os registros são estruturados de forma ordenada, então ele se torna capaz de finalizar o desafio.

Por meio do material concreto, o estudante alcançou o resultado desejado para 2 e 3 discos, porém desistiu ao ser desafiado com mais discos, pois não conseguiu perceber a recursividade. O participante não consegue reconhecer padrões no que diz respeito à ordenação de maior sobre menor, ou ainda mover um disco de cada vez, o que o faz se perder no registro.

A seguir, passamos a analisar a resolução da atividade pelo participante 4.

Figura 37 - Resolução da atividade pelo participante 4



Fonte: resolução do participante 4; produção própria (2020).

Percebe-se que o estudante 4 (figura 37) começa a olhar o problema com atenção e já tentando elaborar estratégias de resolução. Desenha as hastes e os discos para melhor visualizá-los, dividindo a resolução em quatro partes. Esse processo pode ser chamado de decomposição, de acordo com o referencial teórico desta tese. Na primeira parte, ele estabelece o uso de dois, três e quatro discos.

Verifica-se também que o estudante faz uso da habilidade de abstração quando separa as informações importantes que podem ser usadas para a resolução, no caso, a quantidade de discos e até as hastes. Ademais, pode-se dizer que o registro numérico e a estrutura de setas correspondem à realização de uma etapa algorítmica. Essa fase da RP equivale à fase de implementação do plano.

O estudante cria ainda uma estrutura lógica para resolver seu problema ao construir o seu algoritmo, porém não alcança a resposta referente aos cinco discos. Pode-se afirmar que ele realizou a decomposição, isto é, a quebra do número de discos, que, para ele, era o que dificultava a resolução, de modo que a tarefa se tornou possível de se realizar. O estudante faz uso do reconhecimento de padrões quando movimenta as peças sempre levando em consideração a

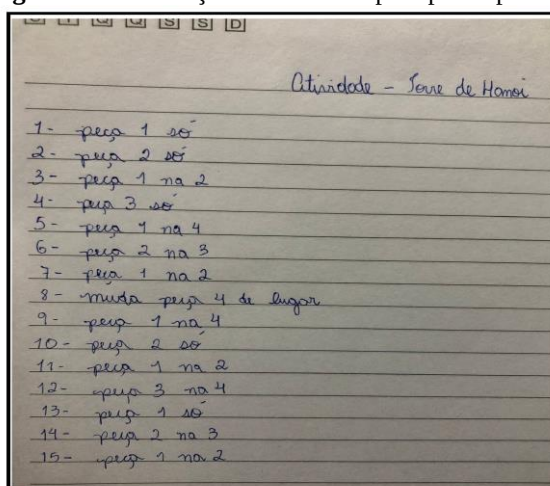
menor sobre a maior. Assim percebe o padrão numérico existente na recorrência dos movimentos e faz uso dessa recursividade, que aumenta de maneira crescente (1, 2, 4, 8...).

Podemos ainda notar que as habilidades de abstração e de reconhecimento de padrões aparecem na hora que o estudante verifica a recorrência, isto é, a repetição de movimentos. Toda vez que o número de discos é par, a execução é a mesma; similarmente, quando é ímpar, é necessária outra sequência. Então, quando o número de discos aumenta, o estudante precisa apenas repetir os passos já executados anteriormente.

Se repararmos no registro numérico do participante — que aqui chamaremos de algoritmo, de acordo com a literatura, pois descreve os passos de seus movimentos —, observamos que, no caso de dois discos, os dois primeiros movimentos simultaneamente são registrados na primeira linha, e logo em seguida o terceiro movimento resolve a situação proposta. No restante do problema, o aluno já reconhece o processo do algoritmo e emprega os mesmos recursos.

Agora, vamos brevemente apresentar a resolução da Torre de Hanói pelo participante 5.

Figura 38 - Resolução da atividade pelo participante 5

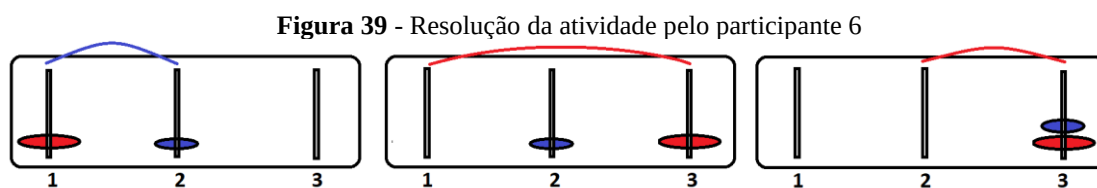


Fonte: resolução do participante 5; produção própria (2020).

Ao analisarmos a RP da figura 38, verificamos que o participante 5 realiza o registro por meio de um passo a passo, escrevendo o seu algoritmo como uma sequência de ações. Seu processo foi mais demorado que o dos demais participantes, pois ele checava cada movimento. É visível, em sua resolução, a construção do pensamento algorítmico, a decomposição, ao dividir sua resolução em etapas; no entanto, ele não consegue reutilizar ações na sua escrita nem criar padrões, embora faça movimentações adequadamente e crie uma estratégia de resolução.

Por fim, o participante 6 demonstra dificuldade na atividade e inicia sua resolução considerando apenas dois discos. O aluno começa movendo o disco menor para a haste 2 e o

disco maior para a haste 3, concluindo que, com dois discos, são necessários três movimentos. Observemos, na Figura 39, o raciocínio desse estudante.



Fonte: resolução do participante; produção própria (2020).

Ao analisar a movimentação (figura 39), percebe-se que o aluno compreendeu a mecânica da atividade e faz uso da decomposição quando decide iniciar o problema com uma quantidade menor de discos para entender a lógica da situação. Porém, ao ser questionado sobre as próximas etapas, disse: “agora não seria só multiplicar por 3 cada disco?”. Nota-se nessa fala um equívoco, já que, quando temos mais discos, os movimentos não são lineares, pois é necessário respeitar a posição de discos maiores e menores, o que implica diferenças na quantidade de movimentos.

Finalizando, podemos observar, nas resoluções dos participantes apresentadas anteriormente, que, ao realizarem a atividade 1, os alunos tiveram a oportunidade de constatar a existência de uma recorrência no processo de resolução.

Verifica-se que o número adicionado é um número parecido com a função 2^n , dessa forma a sequência de números adicionados forma a seguinte progressão geométrica: PG = (2, 4, 8, 16, 32, 63...), cuja razão $q = 2$. Assim, pode-se dizer que 2^{n-1} , logo a quantidade mínima de movimentos é igual ao número somado menos 1.

Os estudantes não conseguiram demonstrar autonomia suficiente para aplicar o padrão observado: para cada incremento de 1 no número de discos, o número mínimo de movimentos dobra, sendo a fórmula geral ($m_x = 2m_{x-1}$) e por progressão aritmético-geométrica ($m_n = 2^n - 1$). Porém, conseguem entender que no processo existe, sim, uma repetição de acontecimentos. Isso é observado na maneira com que registram suas resoluções, utilizando padrões e repetição.

4.3 Processing, aplicação e resultados

Os resultados apresentados nesta seção referem-se às atividades com o *Processing*. Foram realizadas quatro atividades (Quadro 13), sendo que duas foram escolhidas para a análise dos resultados, a escolha das duas atividades para a análise se deu levando em consideração o nível de dificuldade, uma atividade de nível fácil e outra de nível difícil, com maior grau de complexidade na resolução: a calculadora simples e a soma de 0 a 10, nas quais são verificados

os conceitos de decomposição e pensamento algorítmico, o quanto interferem na maneira com que o aluno resolve as atividades e se ele faz uso dessas habilidades em seu processo de resolução.

Relembramos que esta tese objetiva responder às seguintes questões: como ocorre o processo de RP utilizado pelo estudante? Ele aplicaria habilidades do PC conhecida por eles? Quais singularidades dessas habilidades conseguimos perceber em sua resolução? Quais regularidades da decomposição e do pensamento algorítmico percebemos nesse processo da RP?

A seguir, apresentamos as atividades conectadas desenvolvidas pelos alunos durante a aplicação da pesquisa. Os voluntários as realizaram durante três dias consecutivos, em blocos de aula com duração de 2 horas e 40 minutos. O ambiente de programação *Processing* utilizado foi a versão *desktop* do aplicativo.

Quadro 17 - Atividade 1 – Processing – calculadora simples

Atividade 1 Cálculadora Simples	
Problema 1- Processing	
Faça um algoritmo que receba dois valores numéricos e um terceiro valor textual que será definido pelos símbolos: “+”, “-”, “*” e “/”. Dessa forma, o resultado será definido pelo símbolo escolhido. Por exemplo, caso o aluno escolha asterisco (*), será realizada uma multiplicação.	
Descrição	
Com o uso do ambiente de programação, será desenvolvida uma calculadora simples em que o algoritmo receberá dois valores numéricos. O estudante pode escolher a operação que deseja realizar: soma, subtração, multiplicação ou divisão. Ao final, o aplicativo exibirá o resultado do cálculo definido pelo usuário, ou seja, a operação matemática com os dois valores digitados.	
	Habilidade do PC
Algoritmo: sequência de instruções para que o objetivo seja atingido.	
Decomposição: dividir o problema em unidades menores.	
	Habilidade do Componente Curricular de Matemática – BNCC
(EF06MA23) Construir algoritmo para resolver situações passo a passo. (EF06MA04) Construir algoritmo em linguagem natural e 103apresentá-lo por fluxograma que indique a resolução de um problema simples. (EF07MA05) Resolver um mesmo problema utilizando diferentes algoritmos. (EF07MA06) Reconhecer que as resoluções de um grupo de problemas que têm a mesma estrutura pode ser obtidas utilizando os mesmos procedimentos. (EF07MA07) Representar por meio de um fluxograma os passos utilizados para resolver um grupo de problemas. (EF08MA11) Identificar a regularidade de uma sequência numérica recursiva e construir um algoritmo por meio de um fluxograma que permita indicar os números seguintes. (EF08MA09) Resolver e elaborar, com e sem uso de tecnologias, problemas que possam ser representados por equações.	
	Objetivo
Por meio da resolução da atividade, o aluno compreenderá como utilizar os comandos de entrada e saída de dados, bem como compreenderá o processo de entrada, o processamento e a saída de informações que compõem praticamente todo o uso de computação que realizamos, desde as atividades mais simples até mais complexas.	

Fonte: desenvolvido pela autora (2021).

Quadro 18 - Atividade 2 – Processing – verificação do mês do ano

Atividade 2- Verificação de mês do ano	
Problema 2- Processing	
Utilizando o <i>Processing</i>, elabore um algoritmo que receba um número e retorne por extenso o nome do mês do ano. Caso o usuário digite um número inválido, o algoritmo retorna para o usuário que o mês não existe.	
Descrição	
Nesta atividade o estudante compreenderá o conceito de estruturas condicionais, comandos <i>if</i> (se) ou <i>Case</i> (caso), havendo uma variedade de respostas que podem ser obtidas a partir dos dados que o usuário insere. Por exemplo, caso o usuário digite 6, o algoritmo retorna como resultado “Junho”; se o mesmo usuário digitar 15, a aplicação retorna “Mês inválido”.	
 Habilidade do PC	
Algoritmo: sequência de instruções para que o objetivo seja atingido.	
Decomposição: dividir o problema em unidades menores.	
 Habilidade do Componente Curricular de Matemática -BNCC	
(EF06MA04) Construir algoritmo em linguagem natural e 104representá-lo por fluxograma que indique a resolução de um problema simples (por exemplo, se um número natural qualquer é par). (EF06MA23) Construir algoritmo para resolver situações passo a passo (como na construção de dobraduras ou na indicação de deslocamento de um objeto no plano segundo pontos de referência e distâncias fornecidas etc.). (EF06MA26) Resolver problemas que envolvam a noção de ângulo em diferentes contextos e em situações reais, como ângulo de visão.	
 Objetivo	
A atividade tem por objetivo desenvolver o conceito de estrutura condicional, que é amplamente utilizado na resolução de problemas computacionais, pois, quando há várias possibilidades a serem escolhidas, o algoritmo toma a decisão com base nos dados inseridos pelo usuário. Por exemplo: ao digitar um número inteiro, o algoritmo retorna se o valor é par ou ímpar. Assim, caso o resto da divisão de um número dividido por dois seja zero, o valor será par; caso seja 1, o valor será ímpar.	

Fonte: desenvolvido pela autora (2021).

Quadro 19 - Atividade 3 – Processing – soma dos números de 0 a 10

Atividade 3 Soma de números de 0 a 10	
Problema 3 – Processing	
Desenvolva um algoritmo que realize, de forma automática, a soma dos números definidos em um intervalo pelo usuário. Nesse caso o valor inicial é zero e o valor final é dez. Assim, o algoritmo retorna o valor da soma entre esses valores.	
Descrição	
Esta atividade explora o conceito de estrutura de repetição, que pode ser realizada por meio de três comandos em <i>Processing</i>: <i>while</i>, <i>for</i> ou <i>repeat</i> (enquanto, para ou repita). O objetivo é resolver um cálculo que contenha uma estrutura repetitiva de maneira simplificada. No caso exemplificado, o algoritmo é configurado por meio dos laços de repetição para realizar o cálculo enquanto os valores numéricos estiverem no intervalo definido pelo usuário.	
 Habilidade do PC	
- Decomposição. - Algoritmo.	
 Habilidade do Componente Curricular de Matemática – BNCC	
EF08MA04 Resolver e elaborar problemas, envolvendo cálculo, incluindo o uso de tecnologias digitais. EF08MA10 Identificar a regularidade de uma sequência numérica ou figural não recursiva e construir um algoritmo por meio de um fluxograma que permita indicar os números ou as figuras seguintes. EF09MA06 Compreender as funções como relações de dependência unívoca entre duas variáveis e suas representações numérica, algébrica e gráfica e utilizar esse conceito para analisar situações que envolvam relações funcionais entre duas variáveis.	
 Objetivo	
Assim como as estruturas condicionais, as estruturas de repetição também têm grande importância na resolução de algoritmos, pois permitem que cálculos complexos com uma estrutura recursiva possam ser resolvidos de maneira muito mais sintética. Isso faz com que o algoritmo fique bem menor (no que diz respeito à quantidade de linhas de código), bem como promove a sofisticação do entendimento dos conceitos de pensamento algorítmico pelo usuário.	

Fonte: desenvolvido pela autora (2021).

Quadro 20 - Atividade 4 – Processing – Tabuada

Atividade 4 Tabuada	
Problema 4- Processing	
Implemente um algoritmo que receba um número inteiro de 0 a 10 e efetue o cálculo da tabuada desse valor. A tabuada deve realizar a multiplicação do valor digitado por zero até dez.	
Descrição	
O conceito do cálculo da tabuada é introduzido logo no segundo do Ensino Fundamental, e sua utilidade se aplica em diversos conceitos matemáticos importantes que estudamos durante a vida acadêmica. Dessa forma, o aprendizado por meio do algoritmo desenvolvido em um ambiente de programação confere maior profundidade ao entendimento matemático da multiplicação, tanto no que se refere ao pensamento algorítmico quanto à decomposição.	
 Habilidade do PC	
Decomposição. Algoritmo. Abstração: sequência de instruções para que o objetivo seja atingido.	
 Habilidade do Componente Curricular de Matemática – BNCC	
(EF05MA07) Resolver e elaborar problemas de adição e subtração com números naturais e com números racionais, cuja representação decimal seja finita, utilizando estratégias diversas, como cálculo por estimativa, cálculo mental e algoritmos. (EF06MA02) Reconhecer o sistema de numeração decimal, como o que prevaleceu no mundo ocidental, e destacar semelhanças e diferenças com outros sistemas, de modo a sistematizar suas principais características (base, valor posicional e função do zero), utilizando, inclusive, a composição e decomposição de números naturais e números racionais em sua representação decimal.	
 Objetivo	
Desenvolver no estudante, de forma mais aprimorada, os conceitos de laços de repetição (<i>for</i> , <i>while</i> e <i>repeat</i>). Caso o mesmo algoritmo fosse resolvido sem o uso de estruturas de repetição, o código-fonte do algoritmo seria muito mais trabalhoso.	

Fonte: desenvolvido pela autora (2021).

4.3.1 Análise das atividades com o Processing

Os experimentos envolveram a resolução de quatro algoritmos em *Processing*, como detalhado anteriormente. A análise aprofundada dessas resoluções e a coleta de dados focam dois algoritmos. O primeiro, a Atividade 1 (Quadro 17), envolve a resolução de um algoritmo que utiliza somente operadores aritméticos, sendo dessa forma o algoritmo de resolução mais simples da lista.

O segundo algoritmo verificado, a Atividade 3 (Quadro 19), propõe o uso de estruturas de repetição (*for* ou *while*), em que o algoritmo já tem pré-estabelecidos os valores inicial e final para a resolução da conta, ou seja, a soma dos valores numéricos no intervalo entre zero e dez. Ao final, o algoritmo exibe o resultado dessa soma; no caso, o valor da soma é 55.

4.4 Análise das resoluções

Na etapa 4, os participantes da pesquisa resolveram os algoritmos seguindo as instruções e os exemplos explicados durante o minicurso de preparação de uso do *Processing*.

Participante 1
Atividade 1

A resolução a seguir demonstra o que aluno conseguiu produzir com base no enunciado estabelecido. A declaração de variáveis e posteriores cálculos e exibição de respostas estão logicamente corretos. Os erros apresentados na resolução decorrem da falta de experiência com a ferramenta, por exemplo, a ausência do (;) ao final das linhas 3, 4, 5 e 6 da imagem.

A decomposição e o pensamento algorítmico atestam um bom nível de entendimento da proposta, pois o estudante criou as variáveis necessárias e estabeleceu a separação das partes para a resolução de maneira logicamente coerente. As linhas 9 e 10 (figura 40) apresentam erros decorrentes da experimentação dos tipos de variáveis para a resolução da atividade proposta.

Figura 40 - Participante 1: atividade 1

```

mySketch
1  int n1 = 6;
2  int n2 = 8;
3  print(n1, "+", n2, "=", n1 + n2)
4  print(n1, "-", n2, "=", n1 - n2)
5  print(n1, "/", n2, "=", n1 / n2)
6  print(n1, "X", n2, "=", n1 * n2)
7
8
9  dooble
10 string

```

Fonte: desenvolvida pela autora (2021).

Atividade 3

A resposta ao algoritmo dada pelo participante da pesquisa demonstra sucesso em seu entendimento (figura 41). Porém, houve falha na linha 8 da resolução, em que seria necessário que a resposta e a apresentação do resultado da soma ficassem fora da estrutura de repetição (*while*), pois, da forma como foi feito, o algoritmo mostrará 11 vezes o resultado da soma.

Figura 41 - Participante 1: atividade 3

```

1  int soma = 0;
2
3  int cont = 0;
4
5  while (cont <= 10) {
6      soma = soma + cont;
7      cont = cont + 1;
8      println(soma);
9  }

```

Fonte: desenvolvida pela autora (2021).

Participante 2**Atividade 1**

Atestamos frágil entendimento do participante nesse exercício, pois ele emprega conceitos de estruturas condicionais desnecessários para a resolução do algoritmo. Até mesmo a definição de variáveis apresenta problemas em relação ao resultado da divisão, pois a divisão de dois números inteiros geralmente não produz outro número inteiro como resultado.

Nas linhas 7, 9 e 13, o participante pretende apresentar o resultado de um único cálculo; assim concluímos que sua compreensão foi insuficiente ou prejudicada.

Em suma, constatamos o desenvolvimento de um algoritmo mediano nos quesitos decomposição e pensamento algorítmico (figura 42). Tais fragilidades na resolução geralmente são comuns em início de cursos sobre conceitos de programação e, com a sequência do aprendizado, são devidamente sanadas.

Figura 42 - Participante 2: atividade 1

```
1 //calculadora
2 int n1=5
3 int n2=2;
4 int res;
5 { res=n1+n2;
6   println ("a soma é igual a ",res);
7   else
8     res=n1-n2;
9     println ("a subtração é igual a ",res);
10  else
11    res=n1*n2;
12    println ("a multiplicação é igual a ",res);
13  else
14    res=n1/n2;
15    println ("a divisão é igual a ",res);
16 }
```

Fonte: desenvolvida pela autora (2021).

Atividade 3

A estrutura do pensamento algorítmico do estudante está parcialmente correta (figura 43), pois verifica-se certa confusão no uso das variáveis bem como na escolha daquela que armazenará a soma dos valores de forma geral e de qual realizará a contagem até 10. O uso de nomes genéricos de variáveis ocasionou o erro do participante; quando o comentário foi realizado em sua resolução, ele prontamente percebeu o mal-entendido. Em relação à decomposição, a resolução está adequada. Ademais, a quantidade e o uso das variáveis bem como a organização estrutural do algoritmo estão bem-fundamentados.

Figura 43 - Participante 2: atividade 3

```

1 int n1 = 0;
2
3 int n2 = 0;
4
5 while (n2 <= 10) {
6     n2 = n2 + n1;
7     n1 = n1 + 1;
8 }
9 println("A soma no intervalo entre 0 e 10 é igual a ", n2);

```

Fonte: desenvolvida pela autora (2021).

Participante 3

Atividade 1

A resolução apresentada a seguir apresenta poucos problemas e representa uma evolução em relação à proposta inicial, pois basta o algoritmo receber os valores e realizar as quatro operações matemáticas (soma, subtração, multiplicação e divisão). Porém, o estudante cria uma variável que recebe o operador aritmético correspondente ao cálculo, e o algoritmo executa somente a operação matemática definida no valor da variável chamada “operador” (figura 44).

Percebe-se que tanto a decomposição quanto o pensamento algorítmico foram compreendidos de maneira mais sofisticada pelo participante, como consta nas linhas 4, 6, 10, 14 e 18.

Figura 44 - Participante 3: atividade 1

```

1  double n1 = 5;
2  double n2 = 10;
3  double res;
4  string operador = "+"
5
6  else if (operador == "+"){
7      res = n1 + n2;
8      println("O resultado é:", res);
9  }
10 else if (operador == "-"){
11     res = n1 - n2;
12     println("O resultado é:", res);
13 }
14 else if (operador == "*"){
15     res = n1 * n2;
16     println("O resultado é:", res);
17 }
18 else if (operador == "/"){
19     res = n1 / n2;
20     println("O resultado é:", res);
21 }

```

Fonte: desenvolvida pela autora (2021).

Atividade 3

A resposta desenvolvida pelo participante denota uma maneira diferente de se resolver a questão (figura 45), o que é natural em resoluções de algoritmo, que geralmente contemplam diversas possibilidades para se atingir o mesmo resultado. O participante começa o contador da resposta no valor 10 e vai subtraindo até 0, conjuntamente à soma de cada valor. Desse modo, o algoritmo também atinge o resultado esperado. Quando questionado a respeito da sua forma de resolução, o participante comentou que “era mais fácil visualizar a resolução desta maneira”. Ou seja, a decomposição do algoritmo para ele fazia mais sentido partindo desse ponto de vista.

Figura 45 - Participante 3: atividade 3

```

1 int contador = 10;
2
3 int somageral = 0;
4
5 while (contador >= 0) {
6     somageral = somageral + contador;
7     contador = contador - 1;
8 }
9 println(somageral);

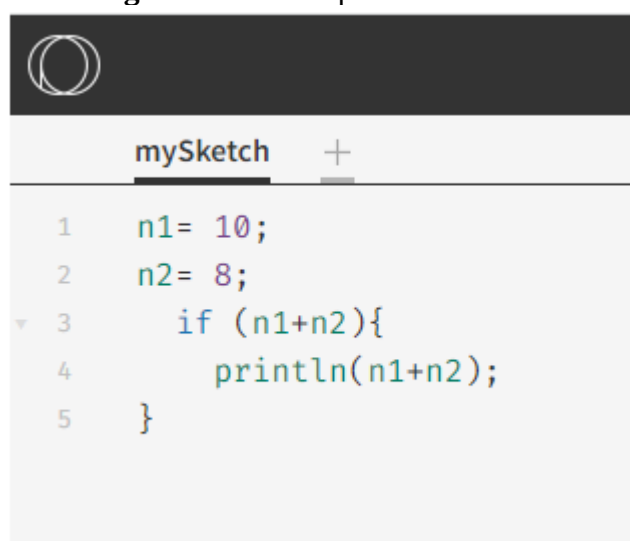
```

Fonte: desenvolvida pela autora (2021).

Participante 4

Atividade 1

A resposta desenvolvida pelo participante ficou incompleta e errada por diversos motivos (figura 46). As variáveis não foram declaradas corretamente, além disso faltou um melhor entendimento sobre os conceitos de variáveis e estruturas condicionais. A resolução indica um estágio inicial de compreensão dos conceitos de decomposição e pensamento algorítmico.

Figura 46 - Participante 4: atividade 1


```

1 n1= 10;
2 n2= 8;
3 if (n1+n2){
4     println(n1+n2);
5 }

```

Fonte: desenvolvida pela autora (2021).

Atividade 3

Essa resolução do participante não apresenta o resultado correto, pois há erros na sintaxe do código, que foram definitivamente corrigidos (figura 47). Aqui cabe a questão de como o participante estruturou o pensamento algorítmico por meio do comando *for* (para), que também é uma estrutura de repetição bastante utilizada e que foi abordada durante o minicurso.

Analisando a atividade especificamente, conclui-se que a compreensão foi satisfatória, pois o estudante claramente sabe o que cada parte do comando *for* realizar. Observando a linha 2, o participante inicia a variável com o valor correto e insere o comando de interrupção da forma certa, o que já constitui um grande avanço na questão tanto da decomposição quanto do pensamento algorítmico.

Figura 47 - Participante 4: atividade 3

```

1  int soma;
2  for (int i= 0; i <= 10, i=i+1) {
3      soma = soma + i;
4  }
5  println(soma);
6

```

Fonte: desenvolvida pela autora (2021).

Participante 5

Atividade 1

Nas linhas 1 e 2 desse exercício, o algoritmo desenvolvido para a resolução apresenta as variáveis e seus tipos, porém o participante conseguiu desenvolver apenas o cálculo da soma, não sendo capaz de evoluir para os demais cálculos propostos (figura 48). Sendo assim, fica implícito que o estudante precisa entender melhor a decomposição para realizar cada cálculo corretamente, bem como compreender, de forma mais adequada, como a entrada, o processamento e a saída de informações ocorrem.

Figura 48 - Participante 5: atividade 1

```
mySketch
1  int n1=5;
2  int n2=5;
3  // soma
4  {
5      println(n1+n2 =res)
```

Fonte: desenvolvida pela autora (2021).

Atividade 3

A resposta desenvolvida por esse participante está correta, pois o algoritmo realmente realiza a soma dos valores entre 0 e 10 e apresenta o resultado correto (figura 49). Porém, no tocante ao pensamento algorítmico, o participante não compreendeu, de maneira correta, o conceito de estruturas de repetição. Percebe-se isso pois foram declaradas 10 variáveis para a execução da soma. Quando confrontado com a seguinte questão: “Por que você não utilizou os comandos *while* ou *for*? Como você faria se o algoritmo pedisse a soma dos valores entre 0 e 1000?”, o participante respondeu: “Putz, ia ficar muito grande e demoraria muito”.

No caso específico, na sequência do término da atividade, foi apresentada ao participante uma resolução envolvendo estruturas de repetição, e sua reação foi de surpresa. Ele disse: “Nossa, fica muito mais fácil e menor assim, mosquei!”.

Figura 49 - Participante 5: atividade 3

```
1 int soma;  
2 int n1 = 1;  
3 int n2 = 2;  
4 int n3 = 3;  
5 int n4 = 4;  
6 int n5 = 5;  
7 int n6 = 6;  
8 int n7 = 7;  
9 int n8 = 8;  
10 int n9 = 9;  
11 int n10 = 10;  
12  
13 soma = n1 + n2 + n3 + n4 + n5 + n6 + n7 + n8 + n9 + n10;  
14  
15 println(soma);
```

Fonte: desenvolvida pela autora (2021).

Participante 6

Atividade 1

O estudante conseguiu compreender, de forma adequada, as estratégias para decomposição do algoritmo. Faltam alguns detalhes no código-fonte para que o algoritmo funcione adequadamente, porém a resolução atesta um adequado conhecimento sobre decomposição e pensamento algorítmico para que a resposta seja correta (figura 50).

Figura 50 - Participante 6: atividade 1

```

mySketch
1  int num=5;
2  //+ , - , * , /;
3  if (num 1 + num 2= res);
4  {
5  println ( "o resultado da soma é" res);
6  }
7  else{
8      if ( num 1,- num 2= res);
9      println ( "o resultado da subtração é" =res);
10 }
11 else{
12     if (num 1 * num 2= res);
13     println ("o resultado da multiplicação é" =res);
14 }
15 else{
16     if ( num 1 / num 2 = res);
17     println ( "o resultado da divisão é" = res);
18 }

```

Fonte: desenvolvida pela autora (2021).

Atividade 3

A resolução elaborada pelo participante demonstra dificuldade em entender os conceitos de variável e estruturas de repetição para a resolução de algoritmos mais complexos (figura 51). Embora o algoritmo alcance a resposta correta, isso foi feito de maneira bem simplória. Ao confrontarmos o estudante com resoluções que envolviam corretamente as variáveis e a estrutura de repetição, ele disse: “Mas resolvi certo, professora, não foi?”.

Para demonstrar que sua resolução estava correta, porém ineficiente, ao término da atividade, construímos uma analogia entre soma e multiplicação: somar 10 vezes o valor 8 nos leva ao resultado 80, porém multiplicar 10 por 8 é uma forma bem mais eficiente para se chegar ao resultado. Dessa forma, o participante demonstrou interesse diferente quando abordado com a Atividade 4 da lista (que não foi analisada, porém cabe ressaltar).

Figura 51 - Participante 6: atividade 3

```

1  int soma = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10;
2
3  println(soma);
4

```

Fonte: desenvolvida pela autora (2021).

Participante 7

Atividade 1

Assim como ocorreu com outros participantes, essa resposta demonstra que a compreensão do conceito de decomposição ficou parcialmente deficiente (figura 52). A resolução apresenta os resultados feitos um a um, ou seja, não houve o entendimento de que o resultado deve apresentar os quatro cálculos executados conjuntamente em um único algoritmo. O participante atingiu um nível considerado intermediário tanto na decomposição quanto no pensamento algorítmico.

Figura 52 - Participante 7: atividade 1

```

//soma
int n1=6;
int n2=6;
int res= (n1+n2);
{
    println("resultado é",res);
} |

```

```

//multiplicação
int n1=6;
int n2=6;
int res= (n1*n2);
{
    println("resultado é",res);
}

```

```
//divisão
int n1=6;
int n2=6;
int res= (n1/n2);
{
    println("resultado é",res);
}
```

```
//subtração
int n1=6;
int n2=6;
int res= (n1-n2);
{
    println("resultado é",res);
}
```

Fonte: desenvolvida pela autora (2021).

Atividade 3

A resolução dessa atividade demonstra boa compreensão pelo estudante das habilidades aqui analisadas, a decomposição e o pensamento algorítmico (figura 53). Esse aluno, diferentemente do participante 2, conseguiu entender o pensamento algorítmico de maneira correta, pois, enquanto aquele confundiu o uso das variáveis contador e soma, este estudante conseguiu separá-las corretamente. Nota-se isso no desenvolvimento das linhas 3 a 5, em que é realizado o cálculo da soma envolvendo a estrutura de repetição. A decomposição se mostra bem-sucedida, pois o participante declarou as variáveis, realizou os cálculos corretamente e exibiu o resultado, seguindo os conceitos de entrada, processamento e saída de dados de maneira eficiente.

Figura 53 - Participante 7: atividade 3

```

1  int n1 = 0;
2  int n2 = 0;
3  while (n1 <= 10){
4      n2 = n2 + n1;
5      n1 = n1 + 1;
6  }
7  println("Soma = ", n2);
8

```

Fonte: desenvolvida pela autora (2021).

Participante 8

Atividade 1

A resposta apresentada a seguir desenvolve, de forma razoavelmente coerente, o cálculo para as quatro operações, embora faltem alguns detalhes relativamente simples para que algoritmo efetue os cálculos sem erros (figura 54). No entanto, percebe-se um déficit de entendimento no que se refere tanto à decomposição quanto ao pensamento algorítmico.

Figura 54 - Participante 8: atividade 1

```

mySketch
1  int num=5;
2  //+,-,*,/;
3  if (num 1 + num 2 = res);
4  println ( "o resultado da soma é" res);
5  }
6  else;
7      if ( num 1,- num2 = res);
8      println ( "o resultado da soma é" =res);
9  }
10 else;
11     If ( num 1 * num 2 =res);
12     println ("o resultado da soma é" = res);
13 }
14 else;
15     if ( num 1 / num 2 = res);
16     println ("o resultado da soma é" = res);
17 }
18

```

Fonte: desenvolvida pela autora (2021).

Atividade 3

A resolução do participante aplica, de maneira eficiente, os conceitos de decomposição, pois estão bem estabelecidos os seus critérios. Todavia, seria necessário ter iniciado a variável *cont* (o contador) com o valor inicial, premissa obrigatória para a estrutura de repetição conseguir exibir o resultado (figura 55). Ao verificar o erro, o participante perguntou: “Olhe, professora, não está dando certo”. Em seguida, declaramos: “Olhe para a declaração das variáveis, pois é nesse ponto que está a resposta para resolver corretamente. Leia atentamente a mensagem de erro”.

O participante não conseguiu chegar à resposta e indagou: “Falta mais uma variável?”. Na sequência respondemos que não e indicamos que o ponto para resolver corretamente o algoritmo era mais simples do que imaginava, pois, em determinados contextos, algumas variáveis devem ter um valor inicial para se chegar ao resultado esperado, caso esse dos contadores utilizados em estruturas de repetição. A mensagem de erro apresenta justamente essa

ausência do valor inicial da variável; traduzindo o erro para português, a mensagem seria algo como “A variável local *cont* pode não ter sido inicializada”.

Figura 55 - Participante 8: atividade 3

```
1 int somavalor;  
2  
3 int cont;  
4  
5 while (cont < 11) {  
6     somavalor = somavalor + cont;  
7     cont = cont + 1;  
8 }  
9  
10 println("Soma = ", somavalor);
```

Fonte: desenvolvida pela autora (2021).

4.5 Resultados da decomposição e algoritmo

A análise dos dados a seguir é apresentada seguindo o objetivo da pesquisa, que visa a entender a decomposição e o pensamento algorítmico nas atividades desenvolvidas e resolvidas pelos participantes.

Na literatura são observadas diferentes formas de decompor problemas, podendo isso ser realizado por sua estrutura, função, sequência ou dependência. Em especial, consideramos aqui a decomposição estrutural, em que ocorre a divisão em subproblemas de modo que cada parte é resolvida separadamente e todas são analisadas da mesma forma; e a sequencial, na qual a ordem das operações é importante para todo o processo de resolução. Concluimos que é difícil afirmar qual maneira de decompor é a mais eficiente, pois, em determinados contextos, cada modelo pode se mostrar ser mais apropriado.

No caso das atividades desenvolvidas nesta tese, a decomposição foi utilizada como um meio de resolver o problema e obter mais informações.

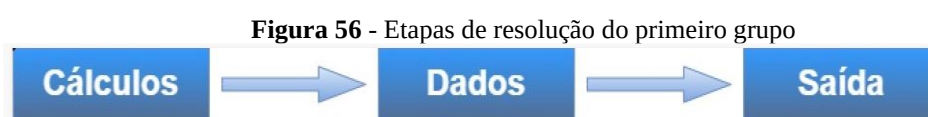
Com base nos dados de resolução e no uso da decomposição, os participantes da pesquisa se dividem em dois grupos:

- Estudantes que decompõem a resolução do problema começando por desenvolver os cálculos que resolverão o algoritmo e ao final mostrarão os resultados pretendidos.

- Estudantes que iniciam primeiramente as mensagens que devem balizar a interação entre usuário e *software*.

No primeiro grupo, verifica-se que há um bom embasamento no conteúdo matemático. Uma vez que os participantes se sentem seguros para resolver o problema, preferem iniciar a resolução do algoritmo em *Processing* pela estrutura matemática já conhecida. Isso demonstra que esse grupo compreendeu melhor a estrutura de decomposição do algoritmo, dando maior ênfase aos cálculos.

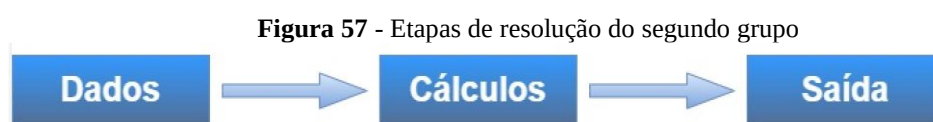
A Figura 56 representa a forma de resolução desses participantes:



Fonte: desenvolvida pela autora (2022).

No segundo grupo estão os estudantes que entenderam melhor a estrutura algorítmica como um todo, considerando a entrada, o processamento e a saída de dados (Figura 11). Os participantes desse grupo iniciam o algoritmo pelas mensagens, pois já compreenderam o fluxo dos dados na resolução de um algoritmo plugado.

A Figura 57 representa a forma de resolução desses participantes:



Fonte: desenvolvida pela autora (2022).

Durante a pesquisa, observou-se que a decomposição auxiliou os participantes a solucionarem os problemas e compreenderem toda a estrutura do problema proposto, ao realizar essa divisão em subproblemas mesmo sem ter uma compreensão global de cada construção realizada. Assim, o participante que só conhece as partes de entrada e saída passa a entender, de forma mais ampla, a função interna do problema, ou seja, o processamento dos dados.

Essa conjectura citada nos parágrafos anteriores é vista como algo positivo, pois contribui, de forma mais profunda, para a resolução de problemas, não se restringindo aos de ordem computacional — afinal o sentido da RP é mais abrangente, estando conectada a disciplinas como matemática, física e química, até mesmo indo além das ciências exatas.

Em relação ao pensamento algorítmico, podemos afirmar, com base nas atividades desenvolvidas nesta pesquisa, que ele consiste na concretização da etapa de decomposição apresentada anteriormente, uma vez que é a partir da maneira como o estudante visualiza a resolução por etapas (decomposição) que o algoritmo é desenvolvido e resolvido.

As atividades desplugadas, em um primeiro momento, ofertaram ao participante maneiras diferenciadas de explorar a resolução de um problema por meio de um algoritmo, por exemplo, utilizar diagramas, fluxogramas ou pseudocódigos (escrever um algoritmo em linguagem natural, como forma de orientar a resolução e a organização). Já os algoritmos resolvidos por meio do ambiente de programação do *Processing* possibilitaram a visão completa do planejamento, execução e resultados (de maneira imediata), pois, por meio da capacidade de processamento dos computadores, os resultados dos cálculos algorítmicos são obtidos instantaneamente.

No desenvolvimento das atividades, foram ressaltadas duas formas de resolução de um algoritmo: (1) focando primeiramente uma estrutura mais simples de resolução (Quadro 17) e (2) depois inserindo gradativamente componentes que impõem maior dificuldade (Quadros 18 a 20) para verificar a compreensão e a análise dos algoritmos pelos participantes.

A princípio o participante deve realizar as operações aritméticas básicas desenvolvendo a compreensão mais simples do que é um algoritmo. Na sequência foram apresentados recursos computacionais (como estruturas condicionais e de repetição), que possibilitam mais flexibilidade na RP, porém impõem maior dificuldade aos participantes. Essa fase é importante para que o aluno entenda cada etapa (assim como a decomposição) e consiga evoluir através das demais estruturas que um algoritmo contém.

Outro aspecto importante diz respeito a estabelecer se houve melhora na compreensão do uso destes dois pilares do pensamento computacional: decomposição e pensamento algorítmico na RP em atividades desplugada e plugada. No Quadro 21 podemos fazer uma analogia entre os dois modelos de atividades desenvolvidos nesta tese. A primeira, a atividade desplugada, serviu como porta de entrada para o conhecimento sobre o PC; já as atividades plugadas no ambiente de programação do *Processing* objetivaram o maior desenvolvimento dessas habilidades.

Quadro 21 - Atividade desplugada e plugada

Desplugado	Plugado (conectado)
Melhora o entendimento da lógica dos estudantes, exibindo o resultado e verificando se está correto ou não.	Melhora o entendimento da lógica dos estudantes, exibindo o resultado e verificando se está correto ou não.

Integra as variáveis em contextos diferentes dos que os estudantes estão habituados.	Amplia o entendimento da matemática, e ela deixa de ser menos teórica.
Antecipa conceitos e experiências que servirão de base para realizar a implementação em ambientes de programação.	Entende o PC como conhecimento que vai além do simplesmente usar, mas que também envolve projetar soluções que resolvem nossos problemas.
	Satisfação imediata com o resultado obtido pelo uso de computadores e ambiente de programação.

Fonte: desenvolvido pela Autora (2022).

No Quadro 21 estão sintetizados alguns aspectos benéficos para ambos os tipos de atividade no momento na RP, pontuando assim algumas habilidades que operam dentro de cada um desses formatos de aplicação, plugado e desplugado.

Ao colocarmos frente a frente duas didáticas diferentes com o mesmo propósito, vemos que essa é uma estratégia eficaz para o desenvolvimento de habilidades específicas do PC. Porém, observa-se que o escopo da atuação das habilidades do PC na RP, quando aplicada apenas a perspectiva desplugada, torna-se menos eficaz. Para o desenvolvimento de habilidades complexas de PC, é necessário o incentivo do uso de atividades plugadas em um ambiente de programação, como o *Processing*.

Os resultados e as análises detalhadas de cada atividade nos mostram que os alunos rapidamente entenderam os conceitos de sequência, repetição e decomposição na realização de atividades simples, como no Quadro 21. As habilidades de abstração, decomposição, reconhecimento de padrão e pensamento algorítmico foram sendo aprimoradas à medida que eram praticadas.

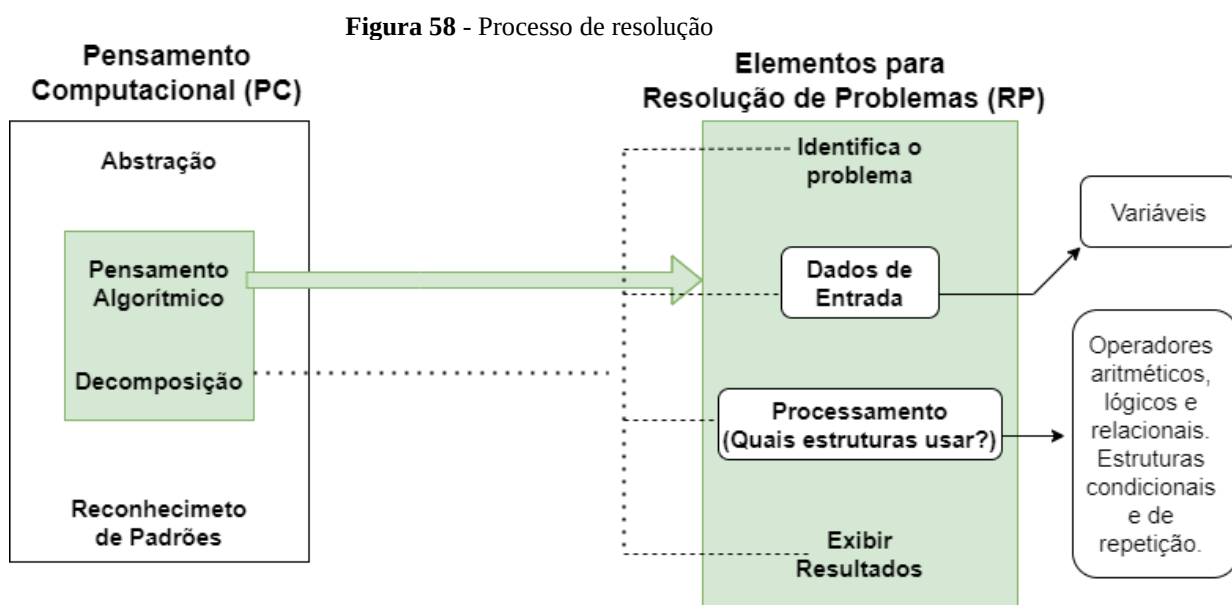
Em contrapartida, alguns participantes não conseguiram decompor corretamente quando o problema era mais sofisticado e depurar os erros em seus códigos. Percebe-se que, se os alunos testassem seus códigos com mais frequência, teriam a chance de corrigir os erros.

Pode-se dizer que os desafios encontrados nos programas implementados em *Processing* geram inúmeras implicações para o ensino de habilidades do PC por meio de programação baseada nos resultados. As implicações incluem considerações instrucionais para decomposição e pensamento lógico.

De forma geral, respondemos às questões do estudo por meio da análise de questionários, vídeos dos desenvolvimentos das atividades laboratoriais que foram gravados e registrado no diário de bordo, da coleta de dados e da análise documental, integrando as habilidades do PC na resolução de problemas matemáticos, sendo elas abstração, reconhecimento de padrões,

pensamento algorítmico e decomposição, com ênfase nos dois últimos e em como eles foram empregados pelos participantes ao realizarem a RP.

A Figura 58 sintetiza o processo desde o seu início do processo até a obtenção final do algoritmo, o que nos permite compreender o processo dos participantes frente às habilidades do PC.



Fonte: desenvolvida pela autora (2022).

Um ponto importante e que exige atenção é a qualidade da compreensão durante o desenvolvimento da programação, visto que ela aconteceu de maneira assertiva, à medida que os alunos vão compreendendo habilidades do PC, lógica e mecanismos da programação, as resoluções passam a ter uma melhor depuração e sofisticação no que diz respeito ao pensamento algorítmico e decomposição. Desta forma conclui-se que quando aplicamos as atividades com raciocínio matemático, aliada a programação, obtemos soluções mais precisas.

Pode-se dizer assim, que é benéfico ensinar PC para RP matemáticos iniciando discussões sobre: habilidades de PC, atividades desplugadas, e atividades plugadas (no caso desta tese- O ambiente de programação Processing).

CAPÍTULO 5 – CONSIDERAÇÕES FINAIS

Nesta tese, em suas aplicações e experimentações, o desempenho não desempenha papel importante. A observação do pensamento algorítmico e da decomposição na resolução das atividades consistiu no nosso maior *corpus*, assim como a forma pela qual os participantes desenvolvem a RP enquanto fazem uso dos pilares do PC.

Quanto aos conhecimentos matemáticos utilizados pelos participantes, percebeu-se que muitas vezes, durante o processo de resolução, eles viram como necessário reformular o problema, passando a utilizar, de forma eficaz, os recursos e conhecimentos matemáticos que já possuíam. Nesse contexto, aplicaram o conhecimento matemático intuitivo, aquele conhecimento inato ou prévio, já adquirido anteriormente, e ainda o conhecimento matemático formal, aprendido durante a educação matemática, como tabuada, razão, proporção, teoremas, entre outros.

O grande desafio para esses participantes foi aprender como empregar os conceitos matemáticos que já possuíam em um contexto diferente e em situações-problemas mais sofisticadas. Por isso, analisar a RP do participante foi importante, para que pudéssemos intervir nesse processo e ajudá-lo a aprimorar questões lógicas e computacionais, minimizando falhas e permitindo que reutilizasse recursos já aprendidos.

Dessa forma, pode-se dizer que o uso de PC para se resolver problemas no âmbito da matemática é eficiente, pois, independentemente do seu grau de complexidade, a evolução na RP é favorecida. Destacamos que o mais importante é que o estudante realize todas as etapas do processo e reaproveite resoluções anteriores, para que prossiga nas suas atividades. Ao observamos a Figura 59, vemos uma sintetização clara de como o mesmo processo pode ser aplicado e as mesmas habilidades (nesse caso, em especial a decomposição e o pensamento algorítmico) podem ser empregadas para a resolução de problemas de diferentes complexidades, claro que em menor ou maior grau.

Cabe aqui retomar Polya (2006), que propõe a RP por meio das seguintes etapas: (1) compreender o problema; (2) elaborar um plano; (3) executar o plano; (4) avaliar a solução. Seguir estratégias como a citada facilita o percurso para a resolução de problemas matemáticos, quando bem compreendido; porém, proporcionar ao estudante a opção de buscar e entender o seu próprio processo de resolução através de habilidades conhecidas dá visibilidade à sua evolução, sejam os problemas simples ou complexos.

É certo que a RP não equivale a simplesmente aplicar conceitos matemáticos; trata-se, na verdade, de uma alternativa positiva para atribuir à matemática seu verdadeiro valor no dia a dia, aprimorando sempre o potencial de interpretação, análise e organização do estudante, antes mesmo que ele aplique conceitos matemáticos.

Nesse sentido, é interessante perceber que obtivemos respostas bastante objetivas e claras, que nos trazem perspectivas bastante sólidas sobre o uso desses processos e ferramentas no ensino de matemática na Educação Básica. Uma de nossas primeiras questões relacionava-se ao processo de *performance* da RP pelo aluno e ao que seria possível observar nesse processo numa aplicação. Ao longo dos percursos da tese e, em especial, em sua experimentação investigativa, com a aplicação de atividades em sala de aula, percebemos os estudantes estimulados a valer-se de suas próprias ferramentas e instrumentos, motivados em busca de uma resolução concreta, quando a aplicabilidade da solução se mostra ao aluno. Nesse cenário a utilização da RP se dá com bastante êxito e em etapas muito claras e bem definidas.

Assim, independentemente de sua complexidade, os problemas apresentados no âmbito da matemática tornam-se concretos, e mesmo as mais intrincadas abstrações ficam facilitadas, uma vez que o aluno se torna capaz de enxergar uma possibilidade de aplicação no mundo que o cerca.

Além disso, o PC, em suas derivações, composições e decomposições, bem como a articulação com a linguagem explicativa e simples, que apresente as informações necessárias ao entendimento claro e preciso do problema, mostraram-se não só presentes, mas essenciais nesse processo de aplicação. Ou seja, os alunos empregaram habilidades de PC especialmente após entender as habilidades necessárias a serem empregadas nos problemas medidos.

A atividade desplugada é uma alternativa positiva e possível para o desenvolvimento inicial das habilidades do PC, e ainda um facilitador para regiões onde o acesso à tecnologia é restrito. Se aliada à RP, o ganho é ainda maior: o estudante passa a compreender o problema com mais propriedade. No entanto, devemos enfatizar que apenas sua aplicação não garante o resultado global benéfico que o PC pode oferecer e ainda limita o estudante a não utilizar ferramentas digitais.

Quando a atividade desplugada passa a ser integrada com a conectada/plugada, percebe-se um resultado mais consistente e positivo tanto no desenvolvimento lógico do participante quanto nas habilidades de PC. Por isso, concluímos que a junção de ambas é mais assertiva e que o uso de tecnologias digitais faz com que o processo de resolução de problemas matemáticos se torne mais claro, dinâmico e menos abstrato para o estudante.

O uso de ambientes de programação permite o aluno coloque em prática conceitos teóricos proporcionando maior envolvimento e participação durante as atividades, sendo assim essencial, no ambiente escolar, uma estrutura tecnológica que possibilite essas oportunidades aos estudantes.

Acompanhando o processo de resolução do aluno com a utilização do ambiente de aprendizagem *Processing*, foi observado como o PC pode interferir na forma de resolver determinados problemas matemáticos e ainda facilitar a sua resolução. Pudemos perceber, assim, que os participantes construíram fortes bases no PC, por meio tanto das atividades desplugadas como daquelas com uso de recursos tecnológicos. Isso fica muito claro e bastante definido quando comparamos os registros realizados nas atividades desplugadas. Brackmann (2017) comprova a eficácia de atividades com abordagem desplugada no ensino do PC, mas, nesta pesquisa, conseguimos perceber que essa eficácia vai além quando considerados os estímulos devidos e trabalhadas as capacidades específicas. Também encontramos uma forte conexão entre RP e as habilidades do PC, com conceitos oriundos da computação — o que por si só gera uma série de desdobramentos possíveis no âmbito educacional, constituindo a computação e seus instrumentos poderosas ferramentas na Educação Básica, mesmo em atividades desplugadas, de modo a propiciar um maior rendimento no ensino de matemática fundamental.

De maneira objetiva, pudemos observar que a decomposição colaborou para reduzir as informações do problema e detalhar sua resolução, de modo que o aluno mantinha o foco apenas em aspectos relevantes para resolver o problema em partes. Além disso, essa divisão do problema facilita a consolidação de conhecimento, pois o estudante consegue visualizar o seu problema de maneira mais abrangente, do início ao fim, o que acarreta uma larga compreensão de toda a situação.

Sabemos que há uma gama bastante variada de trabalhos que medem o PC, contudo a decomposição ainda se configura como uma incógnita em aplicações desse gênero. Justamente por isso esta pesquisa se torna relevante, afinal traz uma série de novas perspectivas nesse sentido: foi possível, com a aplicação e a experimentação deste estudo, perceber que a decomposição pode ser vista como um processo de categorização de elementos, que emprega estratégias diversificadas no momento da execução, e que muitos são os aspectos que ainda podem ser explorados, tais como seus processos, sua categorização e mesmo a comparação entre pessoas que fazem uso dela, como iniciantes e profissionais, o que traz um caráter de aplicação inovador para essa operação e instrumental no ensino de matemática.

Já no que se refere ao pensamento algorítmico, constatamos que ele foi fundamental para que o estudante pudesse estruturar a sua resolução e criar ferramentas para desenvolver e

expressar soluções para problemas, e a análise desse processo veio com as verificações e a investigação sobre as quatro atividades aplicadas. Com base nas atividades desplugadas realizadas, pudemos verificar que o assunto e os resultados são positivos, sendo assim as perspectivas apresentadas devem ser verificadas de maneira aprofundada nos estudos que por ventura provirem desta tese.

Ou seja, temos que os resultados se mostraram bastante satisfatórios, além de apontar que os alunos conseguem resolver problemas com componentes do PC. Um ponto importante e que exige atenção é a qualidade da compreensão durante o desenvolvimento da programação, além de que é importante ressaltar, ainda, que as habilidades do PC podem ocorrer em diferentes tipos de contextos de RP e em outros ambientes de programação.

O impacto desta pesquisa no atual cenário da educação brasileira, tomando em conta seus baixos índices de rendimento no aprendizado de matemática, tem um papel considerável tanto na aplicação diretamente em sala de aula como na formação de professores e coordenadores pedagógicos e, em última instância, até mesmo no desenvolvimento de políticas públicas que possibilitem a instrumentalização desse instrumental em sala de aula.

Outras ferramentas e processos podem ser criados com base nestes conhecimentos e saberes, que podem propiciar a continuidade, a inovação e o desenvolvimento não só de novas pesquisas a partir desta como de estruturas e ferramental para sua utilização nos mais diversos âmbitos educacionais.

O PC é uma ferramenta necessária no século XXI, suas habilidades estão sendo introduzidas nos currículos escolares por isso novos estudos são necessários. Como proposta para trabalhos futuros, deseja-se responder algumas lacunas encontradas e propor abordagens que possibilitariam uma aprendizagem mais assertiva. Como medir o PC em crianças e jovens dos anos iniciais, finais do ensino fundamental e médio? Como aumentar o uso de elementos do PC com infraestrutura educacional existente, para fornecer oportunidades de aprendizado de PC para todos os alunos? Como sequenciar os conteúdos matemáticos e o PC para ser desenvolvido no decorrer do ensino fundamental? Demonstrar como habilidades do PC podem interferir na aprendizagem ao longo de todo ensino fundamental e ensino médio. Além disso, ainda há muita pesquisa necessária para verificar como integrar o PC em outras disciplinas do currículo, beneficiando tanto as áreas específicas quanto a área das Ciências da Computação.

REFERÊNCIAS

- ARAUJO, A. L.; ANDRADE, W.; GUERRERO, D. Um mapeamento sistemático sobre a avaliação do pensamento computacional no Brasil. *In: V CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO*, 2015. **Anais dos Workshops...**, 2016. p. 1.147.
- BAR, V. STEPHENSON, C. Bringing computational thinking to k-12: What is involved and what is the role of the computer science education community? **ACM Inroads**, n. 2, v. 1, p. 48-54, 2011.
- BARCELOS, T. S.; SILVEIRA, I. F. Pensamento computacional e educação matemática: Relações para o ensino de computação na educação básica. *In: XX WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO*, Curitiba, 2012. **Anais do XXXII CSBC**. 2012. p. 23.
- BARCELOS, T. S. *et al.* Relações entre o pensamento computacional e a matemática: uma revisão sistemática da literatura. *In: V CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO*. 2015. **Anais dos Workshops...** 2016. Disponível em: <https://br-ie.org/pub/index.php/wcbie/article/view/6311/4420> Acesso em: 6 nov. 2019.
- BELL, T. *et al.* **Computer science unplugged**: An enrichment and extension programme for primary-aged students. Computer Science Unplugged. 2005.
- BLIKSTEIN, P. **O pensamento computacional e a reinvenção do computador na educação**. 22 dez. 2008. Disponível em: http://www.blikstein.com/paulo/documents/online/ol_pensamento_computacional.html. Acesso em: 27 de jan. 2012.
- BLIKSTEIN, P.; WILENSKY, U. An atom is known by the company it keeps: A constructionist learning environment for materials science using agent-based modeling. **International Journal of Computers for Mathematical Learning**, v. 14, n. 2, p. 81-119, 2009.
- BRACKMANN, C. P. *et al.* Pensamento computacional desplugado: Ensino e avaliação na educação primária espanhola. **Journal on Computational Thinking**, v. 2, n.1, 2018.
- BRACKMANN, C. P. **Pensamento Computacional Brasil**. 2022. Disponível em: <https://www.computacional.com.br/> Acesso em: 30 jan. 2022.
- BRASIL. Ministério da Educação e Cultura. **Parâmetros Curriculares Nacionais: Matemática**. Brasília: MEC/SEF, 1998.
- BRASIL. Ministério da Educação e Cultura. **PCN+ Ensino Médio: Orientações Curriculares Complementares aos Parâmetros Curriculares Nacionais**. Brasília: MEC/SEB, 2002.
- BRASIL. Ministério da Educação e Cultura. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Diretoria de Avaliação da Educação Básica. **Relatório do Sistema de Avaliação Básica (SAEB 2017)**. Brasília: Ministério da Educação, 2019a.
- BRASIL. Ministério da Educação. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Diretoria de Avaliação da Educação Básica. **Relatório Brasil no PISA 2018: versão preliminar**. Brasília: Inep/MEC, 2019b. Disponível em: https://download.inep.gov.br/acoes_internacionais/pisa/documentos/2019/relatorio_PISA_2018_preliminar.pdf. Acesso em: 30 jan. 2022.

CAMPANA, S. B. de C. **Transposição de jogos de tabuleiro utilizados no ensino de Matemática para o formato digital**. 2017. 116 f. Dissertação (Mestrado) - Faculdade de Arquitetura, Artes e Comunicação Universidade Estadual Paulista “Júlio de Mesquita Filho”, Bauru, 2017.

CARVALHO, M. B. de; PENTEADO, M. G. **Informática e educação matemática**. Belo Horizonte: Autêntica, 2019.

CORMEN, T. H. *et al.* **Introduction to algorithms**. Cambridge: MIT Press, 2009.

COSTA, E. J. F.; SAMPAIO, L.; GUERRERO, D. Pensamento computacional na educação básica: Uma análise da relação de questões de matemática com as competências do pensamento computacional. In: V CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO. 2015. **Anais dos Workshops...** 2016.

DA SILVA, F. M.; MENEGHETTI, R. C. G. Matemática e o pensamento computacional: uma análise na pesquisa brasileira. In: XIII ENCONTRO NACIONAL DE EDUCAÇÃO MATEMÁTICA. **Anais...**, 2019.

DANTE, L. R. **Didática da resolução de problemas de matemática**. São Paulo: Editora Ática, 1989.

DICIO, Dicionário Online de Português. Disponível em: <https://www.dicio.com.br/>. Acesso em Novembro, 2022.

DISESSA, A. A. **Changing minds**: Computers, learning, and literacy. Cambridge: MIT Press, 2001.

DISESSA, A. A. Can students re-invent fundamental scientific principles? Evaluating the promise of new-media literacies. In: WILLOUGHBY, T.; WOOD, E. (Eds.) **Children's learning in a digital world**. Oxford: Blackwell, 2007.

DISESSA, A. A. Computational literacy and “the big picture” concerning computers. **Mathematics Education, Mathematical Thinking and Learning**, p. 3-31, 2018.

EUCLIDES. **Os elementos**: Euclides. Tradução e introdução de Irineu Bicudo. São Paulo: Editora UNESP, 2009.

FIORENTINI, D. Alguns modos de ver e conceber o ensino da matemática no Brasil. **Zetetiké**, n. 4, p. 1-37, 1995.

FRANÇA, R. S.; TESDECO, P. C. A. R. Desafios e oportunidades ao ensino do pensamento computacional na educação básica no Brasil. In: V CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2015. **Anais dos Workshops...** 2016. Acesso em: 6 nov. 2019.

GROVER, S.; PEA, R. Computational thinking in K-12: A review of the state of the field. **Educational Researcher**, v. 42, 2013.

KOLODNER, J. L. **Report of a workshop on the pedagogical aspects of computational thinking**. Washington/DC: National Academies Press, 2011.

KONG, S. C.; ABELSON, H. (Eds.). **Computational thinking education**. Springer, 2019.

KWON, K.; CHEON, J. Exploring Problem Decomposition and Program Development through Block-Based Programs. **International Journal of Computer Science Education in Schools**, v. 3, n. 1, p. n1, 2019.

LEE, I. *et al.* Computational thinking for youth in practice. **Acm Inroads**, v. 2, n. 1, p. 32-37, 2011.

LORENCEAU, A.; MAREC, C.; MOSTAFA, T. Upgrading the ICT questionnaire items in PISA 2021. **OECD Education Working Papers**, Paris, n. 202, 2019.

LIUKAS, L. **Hello Ruby: adventures in coding**. New York: Feiwei & Friends, 2015.

MACHADO, E. Z. de A. *et al.* Uma experiência em escolas de ensino médio e fundamental para a descoberta de jovens talento sem computação. In: XVIII WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2010, Belo Horizonte. **Anais do XXX CSBC**, 2010.

MAFRA, J.; ROSSETTI, L. **O diálogo socrático**. Tradução de Janaína Mafra. São Paulo: Paulus, 2015.

MARCHISIO, M. *et al.* Teaching mathematics to non-mathematics majors through problem solving and new technologies. **Education Sciences**, v. 12, n. 1, p. 34, 2022.

MCROBERTS, M. **Arduino básico**. São Paulo: Novatec, 2018.

MESTRE, P. A. A. *et al.* Pensamento computacional: um estudo empírico sobre as questões de matemática do Pisa. In: IV CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2014. **Anais dos Workshops...**, 2015.

MINISTÉRIO DA EDUCAÇÃO. Conselho Nacional de Educação. **Processo n. 23001.001050/2019-18**: Normas sobre computação na educação básica – complemento à BNCC. 2021. Disponível em: http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=182481-texto-referencia-normas-sobre-computacao-na-educacao-basica&category_slug=abril-2021-pdf&Itemid=30192. Acesso em: ago. 2022.

MOORE, G. E. *et al.* Progress in digital integrated electronics. **Internactional Electron Devices Meeting**, p. 11-13, 1975.

MORENO, A. C.; OLIVEIRA, E. Brasil cai em ranking mundial de educação em matemática e ciências; e fica estagnado em leitura. **G1 Educação**, 3 dez. 2019. Disponível em: <https://g1.globo.com/educacao/noticia/2019/12/03/brasil-cai-em-ranking-mundial-de-educacao-em-matematica-e-ciencias-e-fica-estagnado-em-leitura.ghtml>. Acesso em: nov. 2021.

NIEDERAUER, J.; AGUIAR, M. F. C. de. **Desafios e enigmas**. São Paulo: Novera, 2007.

OECD. Programme for International Student Assessment. **PISA 2012 Problem-Solving questions**. 2013. Disponível em: https://www.oecd.org/pisa/pisaproducts/PISA%202012%20framework%20e-book_final.pdf. Acesso em: dez. 2021.

OLIVEIRA, C. M. *et al.* Utilização de desafios para o desenvolvimento do pensamento computacional no ensino superior: um relato de experiência. In: VIII CONGRESSO

BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2019. **Anais do XXX Simpósio Brasileiro de Informática na Educação**, p. 2005.

ONUCHIC, L. de la R. Ensino-aprendizagem de Matemática por meio da resolução de problemas. In: BICUDO, M. A. V. (Org.). **Pesquisa em educação matemática**. São Paulo: Editora UNESP, 1999.

ONUCHIC, L. de la R.; ALLEVATO, N. S. G. Novas reflexões sobre o ensino-aprendizagem de matemática por meio da resolução de problemas. In: BICUDO, M. A. V.; BORBA, M. C. (Orgs.). **Educação matemática: pesquisa em movimento**. São Paulo: Cortez, 2004.

OPEN PROCESSING. Página Inicial. Disponível em: <https://openprocessing.org/>. Acesso em: ago. 2021.

PAPERT, S. **Mindstorms: children, computers, and powerful ideas**. New York: Basic Books, 1980.

PAPERT, S. **Logo: computadores e educação**. Tradução de José Armando Valente, Beatriz Bitelman e Afira V. Ripper. 2. ed. São Paulo: Brasiliense, 1986.

PIAGET, J. **The construction of reality in the child**. New York: Basic Books, 1954.

POLYA, G. **A arte de resolver problemas**. Rio de Janeiro: interciência, 2006.

PROCESSING. Página inicial. [S.d.]. Disponível em: <https://processing.org/>. Acesso em: 7 jan. 2021.

PROCESSING FOR ANDROID. Página inicial. [S.d.]. Disponível em: <https://android.processing.org>. Acesso em: jan. 2021.

PSYCHARIS, S.; KOTZAMPASAKI, E. The impact of a STEM inquiry game learning scenario on computational thinking and computer self-confidence. **Eurasia Journal of Mathematics, Science and Technology Education**, n. 15, v. 4, p. em1689, 2019.

PYTHON MODE FOR PROCESSING. [S.d.]. Disponível em: <https://py.processing.org/>. Acesso em: jan. 2021.

RAABE, A. L. A. *et al.* Referenciais de formação em computação: Educação básica. **Sociedade Brasileira de Computação**, 2017. Disponível em: <https://www.sbc.org.br/files/ComputacaoEducacaoBasica-versaofinal-julho2017.pdf>. Acesso em: 11 jan. 2021.

REAS, C.; FRY, B. **Getting started with Processing: A hands-on introduction to making interactive graphics**. Sebastool: Maker Media, 2015.

RICH, K. M. *et al.* Decomposition: A k-8 computational thinking learning trajectory. In: ACM CONFERENCE ON INTERNATIONAL COMPUTING EDUCATION RESEARCH., 2018. **Proceedings...** 2018. p. 124-132.

RICH, P. J.; *et al.* A framework for decomposition in computational thinking. In: 2019 ACM CONFERENCE ON INNOVATION AND TECHNOLOGY IN COMPUTER SCIENCE EDUCATION, 2019. **Proceedings...** 2019. p. 416-421.

RODRIGUES, R. da S. *et al.* Análise dos efeitos do Pensamento Computacional nas habilidades de estudantes no ensino básico: um estudo sob a perspectiva da programação de computadores. In: XXVI SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO. 2015. **Anais...**, 2015.

ROMANATTO, M. C. Resolução de problemas nas aulas de Matemática. **Revista Eletrônica de Educação**, v. 6, n. 1, p. 299-311, 2012.

SAIBA COMO LAVAR AS ROUPINHAS DO BEBÊ. **Baby Blog**, 15 out. 2012. Disponível em: <https://babypassoapasso.wordpress.com/2012/10/15/saiba-como-lavar-as-roupinhas-do-bebe/>. Acesso em: set. 2019.

SAVIANI, D. **Educação: do senso comum à consciência filosófica**. Campinas: Autores Associados, 2000.

SELBY, C.; WOOLLARD, J. **Computational thinking: the developing definition**. SIGCSE, Atlanta, 5-8 mar., 2013.

SOUZA, E. C. de. Programação no ensino de matemática utilizando *Processing* 2: Um estudo das relações formalizadas por alunos do ensino fundamental com baixo rendimento em matemática. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 11, n. 1, 2018.

TAUB, R.; ARMONI, M.; BEN-ARI, M. CS unplugged and middle-school students' views, attitudes, and intentions regarding CS. **ACM Transactions on Computing Education (TOCE)**, v. 12, n. 2, p. 1-29, 2012.

THE CSTA STANDARDS TASK FORCE. **CSTA K-12 Computer Science Standards**. New York: ACM Computer Science Teachers Association, 2011.

THOMPSON, A G. Learning to teach mathematical problem solving: Changes in teachers' conceptions and beliefs. **The teaching and assessing of mathematical problem solving**, v. 3, p. 232-243, 1989.

TURING, A. M. On computable numbers, with an application to the Entscheidungs problem. **J. of Math**, v. 58, n. 345-363, p. 5, 1936.

UNNIKRISHNAN, R. *et al.* Of elephants and nested loops: How to introduce computing to youth in rural India. **Proceedings of the The 15th International Conference on Interaction Design and Children**. p. 137-146, jun. 2016.

VENTURELLI, S. Software ART/ códigO ARte. In: 9º ENCONTRO INTERNACIONAL DE ARTE E TECNOLOGIA (#9ART). 2010. **Anais...**, Brasília, Instituto de Artes da Universidade de Brasília, 2010.

VAN DE WALLE, J. A. **Matemática no Ensino Fundamental: Formação de professores e aplicação em sala de aula**. Porto Alegre: Penso, 2009.

VIEIRA, A.; PASSOS, O.; BARRETO, R. Um relato de experiência do uso da técnica computação desplugada. In: XXI WEI. 2013. **Anais...**, 2013.

VOLPATO G. L. **Ciência: da filosofia à publicação**. 7. ed. Botucatu: Best Writing, 2019.

VOLPATO, G. L. et al. **Dicionário crítico para redação científica**. 2013.

WING, J. M. Computational thinking. **Commun ACM**, v. 49, n. 3, 2006.

WING, J. M. Computational thinking and thinking about computing. **Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences**, n. 366, v. 1881, p. 3.717-3.725, 2008.

WING, J. M. **Computational thinking**: what and why? 17 nov. 2010. Disponível em: <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>. Acesso em: 6 nov. 2019.

WING, J. M. Computational thinking benefits society. **Social Issues in Computing**, 10 jan. 2014. Disponível em: <http://socialissues.cs.toronto.edu/index.html%3Fp=279.html>. Acesso em: ago. 2022.

WING, J. M. Pensamento Computacional: Um conjunto de atitudes e habilidades que todos, não só cientistas da computação, ficaram ansiosos para aprender e usar. **Revista Brasileira de Ensino, Ciência e Tecnologia (RBECT)**, Ponta Grossa, v. 9, n. 2, p. 1-10, mai./ago. 2016.

APÊNDICE 1 - Iniciativas nacionais e internacionais sobre PC

A presença da tecnologia, de computadores e de dispositivos inteligentes está consolidada na vida diária de todo o mundo. Muitos países, como Inglaterra, Israel, Austrália e Suíça, segundo Relatório da *Computing our Future*, incluíram tecnologia em seus currículos, e mais de 20 países utilizam iniciativas do uso do PC e programação na Educação Básica. Uma das nações precursoras foi o Reino Unido, que transformou os conteúdos de PC em obrigatórios, dando início a eles em 2014, em todos os seus estabelecimentos de ensino da Educação Infantil.

Essas iniciativas de inserção do PC como parte do currículo entre muitos países faz com que nós pesquisadores nos debruçemos sobre o tema. Por isso, estudos sobre PC têm sido objeto de muitos estudos nos últimos anos, pois ele vem sendo caracterizado como uma das habilidades básicas do século XXI, junto com matemática e leitura (PSYCHARIS; KOTZAMPASAKI, 2019).

No Brasil existem algumas instituições que caminham no mesmo sentido, como a Universidade Federal de Pernambuco, Universidade Federal do Rio Grande do Norte e a Univesp (Universidade Virtual do Estado de São Paulo) — esta última inclui PC como sendo uma disciplina básica dos cursos tanto de licenciatura como de bacharelado. Vê-se a necessidade de que o novo profissional que adentrará o mercado de trabalho tenha conhecimento e faça uso dessa habilidade considerada como básica no século em que estamos.

Todos esses estudos e a implementação do PC na Educação Básica e em universidades são ações bem pensadas. Existe uma preocupação em, por meio disso, se fazer ascender outras habilidades, como habilidades lógicas, de linguagem e RP.

Devido a essa demanda de estudos e esforços para a inserção do PC no currículo, muitas organizações apresentam várias iniciativas, gratuitas ou não, para o desenvolvimento dessas habilidades, essenciais na atualidade. Vejamos a seguir algumas dessas iniciativas nacionais e internacionais, que têm como foco a tecnologia e, em especial, o PC.

a) Iniciativa: Sociedade Brasileira de Computação.

Descrição: A SBC possui uma história de mais de 40 anos trabalhando com tópicos relacionados a computação e informática de maneira mais ampla na sociedade, entre eles o PC.

Acesso: <https://www.sbc.org.br/>.

b) Iniciativa: Meninas Digitais.

Descrição: O programa, que tem apoio da SBC, tem por finalidade fomentar o interesse de estudantes do sexo feminino para seguirem nas carreiras que envolvam computação e áreas afins. A iniciativa é interessante, pois a força de trabalho na área de tecnologia é majoritariamente composta por homens. Dessa forma, o programa visa a desmistificar preconceitos sobre o trabalho de mulheres na área.

Acesso: <http://meninas.sbc.org.br/>.

c) Iniciativa: Computação na Escola

Descrição: O projeto nacionalmente está sendo encampado pela Universidade Federal de Santa Catarina (UFSC). Porém, o projeto é internacional e conta com apoio de diversas empresas e personalidades da área de computação. A argumentação principal é que a computação já é umas das principais habilidades do século XXI, portanto não pode ficar de fora dos currículos escolares, pois seu ensino pode impulsionar a inovação e criação de empregos para essa nova era digital.

Acesso: <https://computacaonaescola.ufsc.br/>.

d) Iniciativa: Computing to You (C2Y)

Descrição: A iniciativa é desenvolvida pela Universidade Federal de São Carlos, *campus* de Sorocaba (UFSCar). O projeto, que também possui a parceria com a SBC, visa a introduzir o PC nas escolas da região de Sorocaba/SP. Ele é liderado por professores e estudantes de Ciência da Computação da universidade.

Acesso: <http://uxleris.sor.ufscar.br/c2y/>.

e) Iniciativa: LITE - Univali

Descrição: É um projeto desenvolvido pela Univali (Universidade do Vale do Itajaí), que visa a desenvolver produtos tecnológicos voltados para a área da educação. O espaço do projeto está aberto para pesquisadores desenvolverem seus projetos nos mais diferentes âmbitos do ensino.

Acesso: <http://lite.acad.univali.br/pt/>.

f) Iniciativa: Code.org

Descrição: É uma organização sem fins lucrativos que tem por objetivo incentivar o aprendizado de ciência da computação nas escolas. Incentiva que meninas e outras minorias que não possuam oportunidades tenham a possibilidade de aprender por computação. É um projeto apoiado por empresas como Microsoft e Amazon.

Acesso: <https://code.org/>.

g) Iniciativa: Scratch

Descrição: O projeto Scratch foi desenvolvido pelo MIT (Massachusetts Institute of Technology), universidade norte-americana. O Scratch é um *software* que permite a programação sem a necessidade de codificar de forma textual, que é a usada na maioria das linguagens de programação. A programação é feita em blocos visuais, sendo uma excelente introdução ao entendimento do pensamento computacional.

Acesso: <https://scratch.mit.edu/>.

h) Iniciativa: CS First

Descrição: Embora não exista versão na língua portuguesa, é um projeto interessante, pois tem por objetivo capacitar professores para que sejam disseminadores dos conceitos de ciência da computação, afinal as habilidades da tecnologia serão essenciais para os estudantes em seu futuro profissional e acadêmico.

Acesso: <https://csfirst.withgoogle.com/s/en/home>.

i) Iniciativa: Barefoot

Descrição: Também é um projeto que não está disponível em português e possui a finalidade de aumentar os conhecimentos de computação dos professores do ensino fundamental. É apoiado por professores e por pesquisas.

Acesso: <https://www.barefootcomputing.org/>.

j) Iniciativa: Computing at School (CAS)

Descrição: Tem como incentivadora a autora Jeannette Wing, referência dos estudos sobre o pensamento computacional.

Acesso: <https://www.computingschool.org.uk/>.

k) Iniciativa: Cs Unplugged

Descrição: O projeto visa a apresentar a computação primeiramente de forma desconectada, ou seja, sem computadores, por meio de jogos e outras atividades. Estas últimas também são utilizadas fora do contexto escolar, como em palestras para idosos e outros eventos. Disponível somente em inglês.

Acesso: <https://csunplugged.org/en/>.

l) Iniciativa: Khan-Academy

Descrição: É um projeto mundialmente conhecido que disponibiliza gratuitamente vídeos de diversas áreas do conhecimento, como matemática, medicina, química e computação. A instituição é uma entidade sem fins lucrativos que visa a oferecer maior conhecimento aos estudantes por meio das aulas em vídeo.

Acesso: <https://pt.khanacademy.org/>.

m) Iniciativa: Hello Ruby

Descrição: A iniciativa acredita que saber programar é a alfabetização do século XXI. Dessa forma, o projeto ensina os fundamentos da computação e do pensamento computacional por meio de um livro dividido em tópicos para não apenas ensinar uma linguagem específica, mas também para que as crianças entendam as estruturas presentes em todas as linguagens de programação.

Acesso: <https://www.helloruby.com/>