



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CAMPUS DE RUSSAS**  
**CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**MATHIAS AUGUSTO DE MELO**

**O USO DO OCTOSTUDIO PARA O ENSINO DE PENSAMENTO COMPUTACIONAL  
NO ENSINO FUNDAMENTAL**

**RUSSAS**

**2026**

MATHIAS AUGUSTO DE MELO

O USO DO OCTOSTUDIO PARA O ENSINO DE PENSAMENTO COMPUTACIONAL NO  
ENSINO FUNDAMENTAL

Trabalho de Conclusão de Curso apresentado ao  
Curso de Graduação em Ciência da Computação  
do Campus de Russas da Universidade Federal  
do Ceará, como requisito parcial à obtenção do  
grau de bacharel em Ciência da Computação.

Orientador: Prof. Dr. Markos Oliveira  
Freitas

RUSSAS

2026

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Sistema de Bibliotecas  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

M486u Melo, Mathias Augusto de.  
O USO DO OCTOSTUDIO PARA O ENSINO DE PENSAMENTO COMPUTACIONAL NO  
ENSINO FUNDAMENTAL / Mathias Augusto de Melo. – 2026.  
102 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas,  
Curso de Ciência da Computação, Russas, 2026.  
Orientação: Prof. Dr. Markos Oliveira Freitas.

1. OctoStudio. 2. Pensamento Computacional. 3. Ensino Básico. I. Título.

CDD 005

---

MATHIAS AUGUSTO DE MELO

O USO DO OCTOSTUDIO PARA O ENSINO DE PENSAMENTO COMPUTACIONAL NO  
ENSINO FUNDAMENTAL

Trabalho de Conclusão de Curso apresentado ao  
Curso de Graduação em Ciência da Computação  
do Campus de Russas da Universidade Federal  
do Ceará, como requisito parcial à obtenção do  
grau de bacharel em Ciência da Computação.

Aprovada em: 16/01/2026

BANCA EXAMINADORA

---

Prof. Dr. Markos Oliveira Freitas (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Rafael Fernandes Ivo  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Arnaldo Barreto Vila Nova  
Universidade Federal do Ceará (UFC)

## RESUMO

As Normas sobre Computação na Educação Básica (Ministério da Educação, 2022b) trazem as diretrizes gerais para a inclusão do ensino de Computação no Brasil. Dentre os desafios para a implementação do ensino desta área, está a falta de acesso a computadores por alunos de escolas públicas para a prática dos conteúdos propostos. Uma forma de solucionar este problema é utilizar o OctoStudio, uma aplicação de ensino de programação disponível para dispositivos móveis, como *smartphones* e *tablets*, que são mais acessíveis para estes alunos. Assim, esta pesquisa investiga a eficácia do uso do aplicativo OctoStudio para o ensino de Pensamento Computacional, alinhado à disciplina de matemática, em uma turma do sexto ano do ensino fundamental de uma escola pública. Para isso, este trabalho descreve a realização de um curso, estruturado conforme as sugestões das Normas sobre Computação na Educação Básica, com o objetivo de trabalhar habilidades do eixo de Pensamento Computacional alinhadas ao conteúdo da disciplina de matemática. Para a coleta de dados, foram aplicados questionários para identificar o perfil dos alunos, especialmente seu acesso a computador, à internet e a *smartphone* e seus conhecimentos prévios sobre computação. Também foram identificados, através dos questionários, a satisfação dos alunos com o uso do OctoStudio e os conhecimentos adquiridos após o curso, com o objetivo de analisar o método e a ferramenta aplicada. Os resultados indicam que o OctoStudio foi avaliado pelos alunos do curso como facilitador de aprendizado e divertido, e que a maioria dos alunos relatou que conseguiu programar e que tem interesse em aprender mais sobre programação.

**Palavras-chave:** OctoStudio; pensamento computacional; ensino básico.

## ABSTRACT

The Standards on Computing in Basic Education (Ministério da Educação, 2022b) provide general guidelines for the inclusion of computer science education in Brazil. Among the challenges for implementing this area of education is the lack of access to computers for students in public schools to practice the proposed content. One way to solve this problem is to use OctoStudio, a programming teaching application available for mobile devices such as smartphones and tablets, which are more accessible to these students. Thus, this research investigates the effectiveness of using the OctoStudio application for teaching Computational Thinking, aligned with the mathematics discipline, in a sixth-grade class in a public elementary school. To this end, this work describes the implementation of a course, structured according to the suggestions of the Standards on Computing in Basic Education, with the objective of working on skills in the Computational Thinking axis aligned with the content of the mathematics discipline. For data collection, questionnaires were applied to identify the students' profile, especially their access to computers, the internet, and smartphones, and their prior knowledge of computing. The questionnaires also identified student satisfaction with the use of OctoStudio and the knowledge acquired after the course, with the aim of analyzing the method and the tool applied. The results indicate that OctoStudio was evaluated by the course students as a learning facilitator and fun, and that most students reported that they were able to program and that they are interested in learning more about programming.

**Keywords:** OctoStudio; computational thinking; basic education.

## LISTA DE FIGURAS

Figura 1 – Conceitos do eixo Pensamento Computacional no Ensino Fundamental . . .	18
Figura 2 – Conceitos do eixo Mundo Digital no Ensino Fundamental . . . . .	19
Figura 3 – Conceitos do eixo Cultura Digital no Ensino Fundamental . . . . .	19
Figura 4 – Exemplo de algoritmo implementado com blocos no OctoStudio. . . . .	21
Figura 5 – Página de criação de projetos do OctoStudio . . . . .	23
Figura 6 – Alunos fazendo as suas criações artísticas . . . . .	32
Figura 7 – Criações artísticas dos alunos . . . . .	32
Figura 8 – Projeto criado por uma dupla . . . . .	33
Figura 9 – Amostras dos cartões de início do projeto . . . . .	35
Figura 10 – Atividade emita um sinal . . . . .	43
Figura 11 – Comando Se-então-senão no OctoStudio . . . . .	44
Figura 12 – Primeiro exercício da aula de comandos de controle . . . . .	45
Figura 13 – Segundo exercício da aula de comandos de controle . . . . .	45
Figura 14 – Primeiro exercício da aula de comandos de repetição . . . . .	46
Figura 15 – Segundo e terceiro exercício da aula de comandos de repetição . . . . .	46
Figura 16 – Primeira sugestão de atividade de avaliação . . . . .	47
Figura 17 – Segunda sugestão de atividade de avaliação . . . . .	47
Figura 18 – Jogo de gato que pega estrelas feito por um aluno . . . . .	48
Figura 19 – Programa de movimentação de um porco-espinho feito por um aluno . . . . .	49
Figura 20 – Jogos de instrumentos e chute a gol feitos por alunos . . . . .	49
Figura 21 – Jogo de pegar comidas com panda e movimentação por cenários feito por um aluno . . . . .	50
Figura 22 – Jogo de saltar sobre obstáculos feito por um aluno . . . . .	50
Figura 23 – Resultados das questões sociodemográficas . . . . .	53
Figura 24 – Alunos que sabem o que é criar um programa de computador . . . . .	54
Figura 25 – Opinião dos alunos sobre programar . . . . .	54
Figura 26 – Resultados sobre saber programar e ter tido aulas de programação . . . . .	55
Figura 27 – Respostas sobre a utilidade da computação e seu interesse em trabalhar com ela	55
Figura 28 – Quantidade de acesso semanal e tipo de uso da internet . . . . .	57
Figura 29 – Quantidade de acesso semanal e tipo de uso do <i>smartphone</i> . . . . .	57
Figura 30 – Alunos que usam computador . . . . .	58

Figura 31 – Quantidade de acessos semanais e tipo de uso computador . . . . .	59
Figura 32 – Avaliação dos alunos sobre a qualidade do curso . . . . .	60
Figura 33 – Avaliação dos alunos sobre a dificuldade e o grau de diversão das aulas . . .	61
Figura 34 – Percepção dos alunos sobre o tempo das aulas . . . . .	61
Figura 35 – Avaliação dos alunos sobre saber e gostar de programar . . . . .	64
Figura 36 – Alunos que compartilharam e explicaram seus programas . . . . .	65
Figura 37 – Avaliação dos alunos sobre a dificuldade e o grau de diversão das aulas . . .	66
Figura 38 – Avaliação dos alunos sobre a dificuldade e o grau de diversão de programar .	66
Figura 39 – Respostas sobre a utilidade da computação e seu interesse em trabalhar com ela	67
Figura 40 – Exemplo de código “Bom” . . . . .	69
Figura 41 – Exemplo de código “Razoável” . . . . .	70
Figura 42 – Avaliação da habilidade de “Elaborar algoritmos” . . . . .	70
Figura 43 – Avaliação da habilidade de “Descrever a solução de um problema” . . . . .	71
Figura 44 – Avaliação da habilidade de “Construir soluções automatizadas com decompo- sição” . . . . .	71

## LISTA DE ABREVIATURAS E SIGLAS

BNCC	Base Nacional Comum Curricular
IBGE	Instituto Brasileiro de Geografia e Estatística
LLK	<i>Lifelong Kindergarte</i>
MIT	<i>Massachusetts Institute of Technology</i>
SBC	Sociedade Brasileira de Computação
SOL	SBC OpenLib

## SUMÁRIO

1	<b>INTRODUÇÃO</b> . . . . .	12
1.1	<b>Contextualização</b> . . . . .	12
1.2	<b>Objetivos</b> . . . . .	13
1.2.1	<i>Objetivo geral</i> . . . . .	13
1.2.2	<i>Objetivos específicos</i> . . . . .	14
1.3	<b>Organização do trabalho</b> . . . . .	14
2	<b>FUNDAMENTAÇÃO TEÓRICA</b> . . . . .	15
2.1	<b>Introdução</b> . . . . .	15
2.2	<b>Pensamento Computacional</b> . . . . .	15
2.3	<b>Diretrizes para ensino de Computação na Educação Básica</b> . . . . .	17
2.4	<b>OctoStudio</b> . . . . .	20
2.5	<b>Modelo de avaliação dTECT</b> . . . . .	23
2.6	<b>Síntese do capítulo</b> . . . . .	24
3	<b>REVISÃO BIBLIOGRÁFICA</b> . . . . .	25
3.1	<b>Introdução</b> . . . . .	25
3.2	<b>Metodologia da pesquisa</b> . . . . .	25
3.2.1	<i>Identificação da problemática</i> . . . . .	25
3.2.2	<i>Escolha do repositório</i> . . . . .	26
3.2.3	<i>Estabelecimento de critérios de inclusão/exclusão</i> . . . . .	26
3.2.4	<i>Caracterização dos trabalhos</i> . . . . .	26
3.2.5	<i>Análise dos trabalhos e discussão dos dados</i> . . . . .	26
3.2.6	<i>Resultado da análise</i> . . . . .	27
3.2.7	<i>Apresentação das conclusões</i> . . . . .	27
3.3	<b>Trabalhos relacionados</b> . . . . .	27
3.3.1	<i>Estimulando o pensamento computacional no ensino médio utilizando o software Scratch</i> . . . . .	27
3.3.2	<i>Computação Criativa com OctoStudio no Ensino Fundamental</i> . . . . .	29
3.3.3	<i>The Combination of Programming Language with Artistic Language as a Teaching Resource to Develop Creativity and Digital Competence</i> . . . . .	31
3.3.4	<i>Playful Occupations: Mobile Creative Coding for Critical Consciousness</i> . . . . .	34

3.3.5	<i>Análise comparativa</i> . . . . .	36
3.4	<b>Considerações finais</b> . . . . .	36
4	<b>METODOLOGIA</b> . . . . .	37
4.1	<b>Introdução</b> . . . . .	37
4.2	<b>Aspectos éticos da pesquisa</b> . . . . .	37
4.3	<b>Ementa e desenvolvimento das aulas</b> . . . . .	38
4.3.1	<i>Cronograma das aulas</i> . . . . .	38
4.3.2	<i>Metodologias das aulas</i> . . . . .	39
4.3.3	<i>Materiais utilizados</i> . . . . .	39
4.3.4	<i>Avaliação dos conhecimentos</i> . . . . .	40
4.4	<b>Desenvolvimento dos questionários de pesquisa</b> . . . . .	41
4.5	<b>Aplicação das aulas</b> . . . . .	41
4.5.1	<i>Contextualização da escola e da turma</i> . . . . .	41
4.5.2	<i>Descrição das aulas</i> . . . . .	41
4.5.2.1	<i>Aula 01 - dia 23 de outubro de 2025</i> . . . . .	42
4.5.2.2	<i>Aula 02 - dia 30 de outubro de 2025</i> . . . . .	43
4.5.2.3	<i>Aula 03 - dia 06 de novembro de 2025</i> . . . . .	45
4.5.2.4	<i>Aula 04 - dia 18 de novembro de 2025</i> . . . . .	48
4.6	<b>Exemplos de atividades dos alunos</b> . . . . .	48
4.7	<b>Considerações finais</b> . . . . .	51
5	<b>RESULTADOS</b> . . . . .	52
5.1	<b>Introdução</b> . . . . .	52
5.2	<b>Questionário inicial</b> . . . . .	52
5.2.1	<i>Questões sociodemográficas</i> . . . . .	53
5.2.2	<i>Opiniões e experiências anteriores</i> . . . . .	53
5.2.3	<i>Acesso à internet e tecnologia</i> . . . . .	56
5.3	<b>Questionário final</b> . . . . .	59
5.3.1	<i>Questões sobre as aulas</i> . . . . .	60
5.3.2	<i>Questões sobre aprendizado e envolvimento</i> . . . . .	64
5.3.3	<i>Questões sobre programação e computação</i> . . . . .	66
5.4	<b>Avaliação dos conhecimentos</b> . . . . .	68
5.5	<b>Considerações</b> . . . . .	72

<b>6</b>	<b>CONCLUSÃO</b> . . . . .	<b>73</b>
<b>6.1</b>	<b>Principais contribuições</b> . . . . .	<b>73</b>
<b>6.2</b>	<b>Trabalhos futuros</b> . . . . .	<b>74</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>75</b>
	<b>APÊNDICES</b> . . . . .	<b>78</b>
	<b>APÊNDICE A – TERMO DE CONSENTIMENTO LIVRE E ESCLA- RECIDO</b> . . . . .	<b>78</b>
	<b>APÊNDICE B – TERMO DE ASSENTIMENTO</b> . . . . .	<b>80</b>
	<b>APÊNDICE C – SLIDES DAS AULAS</b> . . . . .	<b>82</b>
	<b>ANEXOS</b> . . . . .	<b>88</b>
	<b>ANEXO A – CONTEÚDOS E HABILIDADES DE COMPUTAÇÃO PARA O 6º ANO - EIXO DE PENSAMENTO COMPU- TACIONAL</b> . . . . .	<b>88</b>
	<b>ANEXO B – TABELA DE QUESTÕES DO DETECT</b> . . . . .	<b>91</b>
	<b>ANEXO C – GUIA DE REFERÊNCIA DO OCTOSTUDIO</b> . . . . .	<b>92</b>
	<b>ANEXO D – CARTÕES DE PROGRAMAÇÃO DO OCTOSTUDIO</b> .	<b>99</b>

# 1 INTRODUÇÃO

## 1.1 Contextualização

Em nosso cotidiano, a computação está presente em praticamente todos os serviços essenciais, do lar ao trabalho, na saúde, agricultura, automóveis e na crescente automação que impõe desafios sociais e econômicos. A informação humana é majoritariamente digital, tornando o mundo cada vez mais dependente das tecnologias da informação e comunicação.

Para conhecer as tecnologias computacionais de forma crítica, ética, segura e eficiente, é preciso compreender o universo digital e suas ferramentas, já que mesmo soluções locais requerem abordagens intersetoriais baseadas no crescente uso de artefatos digitais e conhecimentos cada vez mais interdisciplinares das Ciências, Humanidades e Artes. Nesse contexto, o desenvolvimento das competências educacionais modernas passa inevitavelmente pela Computação.

O avanço da computação transforma não apenas as cadeias produtivas, mas também as relações sociais, as artes e suas formas de criação e apreciação, além das maneiras de aprender e educar. Nesse contexto, o “pensamento computacional” refere-se a um conjunto de habilidades cognitivas voltadas para compreender, modelar, comparar, solucionar e automatizar problemas de forma sistemática, por meio de algoritmos que descrevem com precisão processos complexos, etapas, recursos e informações. Hoje, esse pensamento é reconhecido como uma competência essencial para o século XXI (Ministério da Educação, 2022b).

Dadas as mudanças em curso na sociedade graças aos avanços tecnológicos, métodos para a incorporação do ensino da computação às práticas pedagógicas têm sido buscados ao longo de anos no país, como mostra Ministério da Educação (2022b).

Raabe *et al.* (2020), apresenta como abordagens propostas de introdução da Computação na Educação Básica: 1) Construcionismo e Letramento Computacional; 2) Pensamento Computacional; 3) Demandas do Mercado; e 4) Equidade e Inclusão. Todas estas abordagens buscam ampliar o conhecimento dos estudantes acerca do potencial do computador para resolver problemas, e utilizam o termo pensamento computacional para simbolizar as habilidades cognitivas que estão associadas a programação, desenvolvimento de algoritmos e resolução de problemas

A abordagem de Pensamento Computacional, surge de uma cultura computacional em que cientistas da computação percebem sua relevância para a sociedade e busca a caracte-

rização da computação como área de conhecimento, organizada como uma disciplina que tem conteúdos próprios, para que os estudantes possam compreender o que é programação e como abordar a resolução de problemas computacionais.

A futura inserção profissional de nossos discentes não pode se apoiar apenas em habilidades de uso e consumo de tecnologias digitais, sendo essencial a capacidade de criar. Para isso, é indispensável compreender os fundamentos que sustentam essas tecnologias. Inclusive, a necessária reflexão crítica sobre questões éticas e os impactos sociais do uso da tecnologia exige o domínio dos fundamentos da Computação.

Porém, a infraestrutura escolar e domiciliar, é um dos desafios encontrados para a implantação das práticas de ensino. A disponibilidade de computadores para os alunos foi identificada como um fator influente no aprendizado de programação e Pensamento Computacional pelos trabalhos de Xisto (2023) e Pinho (2024).

O Anuário Brasileiro da Educação Básica de 2024 (Todos Pela Educação, 2024) mostra que apenas 44,7% das escolas do Brasil disponibilizam computadores de mesa para uso dos alunos e somente 29,7% têm laboratório de informática. Além disso, o panorama do Instituto Brasileiro de Geografia e Estatística (IBGE) mostra, em IBGE (2021), que apenas 42,6% dos domicílios no Brasil possuíam microcomputadores ou *tablets*, enquanto 96,3% possuíam telefone móvel celular.

Neste cenário, surge a necessidade de uma aplicação que supra a falta de computadores para os alunos, sendo utilizável em dispositivos de fácil acesso, como é o *smartphone*. Cumprir este requisito é uma das propostas do aplicativo OctoStudio (Rusk *et al.*, 2024).

Por isso, é de suma importância para este trabalho entender quais estratégias têm sido aplicadas em práticas que fazem uso do OctoStudio como ferramenta de ensino, quais conteúdos têm sido desenvolvidos e relacionados ao Pensamento Computacional em atividades com o OctoStudio e quais métodos têm sido aplicados para avaliar a aquisição do conteúdo pelos alunos e avaliar as práticas realizadas.

## 1.2 Objetivos

### 1.2.1 *Objetivo geral*

Este trabalho tem como objetivo principal avaliar o uso do aplicativo OctoStudio como ferramenta para o ensino de Pensamento Computacional para uma turma do 6º ano

do ensino fundamental de uma escola pública. Mais detalhes sobre a escola e a turma são apresentados na subseção 4.5.1.

### **1.2.2 *Objetivos específicos***

Para se chegar ao objetivo geral, foram definidos como objetivos específicos os seguintes:

- Desenvolver uma metodologia de aplicação: elaborar um curso de curta duração utilizando o OctoStudio para o ensino de habilidade de Pensamento Computacional e programação alinhado à disciplina de matemática;
- Selecionar um artefato de avaliação: designar uma ferramenta para avaliação da qualidade do curso elaborado;
- Elaborar um questionário de perfil: catalogar, por meio do questionário, o acesso dos alunos à *smartphones* e internet e seus conhecimentos prévios de programação;
- Indicar desafios e oportunidades: identificar limitações, mapear os principais desafios enfrentados e documentar oportunidades de melhoria, com base nos resultados e observações ao longo do curso.

### **1.3 Organização do trabalho**

Este trabalho está organizado em seis seções, distribuídos de forma a apresentar de maneira lógica e progressiva os elementos constituintes do estudo. A Seção 2 apresenta a fundamentação teórica, abordando os principais conceitos relacionados ao pensamento computacional, à computação na educação básica, ao OctoStudio e ao modelo de avaliação usado nesta pesquisa, o DETECT.

Na Seção 3, são explorados os trabalhos relacionados, destacando pesquisas similares e contribuições anteriores sobre o uso de ferramentas tecnológicas na promoção do pensamento computacional. A Seção 4 descreve a metodologia utilizada, incluindo os procedimentos a serem adotados, os instrumentos de coleta de dados, os critérios de análise e exemplos de programas.

A Seção 5 apresenta os resultados obtidos com a pesquisa, gráficos sintetizando as informações e análises dos dados levantados. Por fim, na Seção 6 estão as conclusões do trabalho com base nos dados e análises desta pesquisa, além dos desafios e limitações do trabalho e sugestões de trabalhos futuros.

## **2 FUNDAMENTAÇÃO TEÓRICA**

### **2.1 Introdução**

Esta seção apresenta os temas que embasam esta pesquisa, com o propósito de tornar claros os seus fundamentos. A Seção 2.2 discorre sobre Pensamento Computacional. Em seguida, a Seção 2.3 traz uma síntese a respeito das diretrizes do Brasil para o ensino de Computação na educação básica. Na Seção 2.4, está uma descrição sobre o aplicativo OctoStudio. Na Seção 2.5, está disposta uma explicação acerca do modelo de avaliação DETECT.

### **2.2 Pensamento Computacional**

O conceito de Pensamento Computacional surgiu com o trabalho de Papert (1980), que menciona o termo, porém não o define. Papert defende que a presença de um computador pode mudar a forma como as pessoas desenvolvem aprendizagens e construir suas próprias conexões e estruturas intelectuais. Este pensamento poderia ocorrer mesmo na ausência da máquina, influenciando o aprendizado em outras áreas do conhecimento.

Apesar de o conceito ter sido utilizado anteriormente, o termo Pensamento Computacional popularizou-se com o artigo de Wing (2006), que define o conceito como um processo de modelar um problema e suas soluções de maneira que possam ser processados por um agente de informação. A autora afirma que o Pensamento Computacional é amplamente aplicável à compreensão do comportamento humano e da habilidade de resolução de problemas, sendo uma habilidade necessária a todos para a competência na vida cotidiana, não apenas para cientistas da computação e programadores.

As Normas sobre Computação da Educação Básica (Ministério da Educação, 2022b) definem o Pensamento Computacional como "o conjunto de habilidades cognitivas para compreender, definir, modelar, comparar, solucionar, automatizar e analisar problemas e possíveis soluções de forma metódica e sistemática por meio de algoritmos que são descrições abstratas e precisas de um raciocínio complexo, compreendendo etapas, recursos e informações envolvidos num dado processo".

O mesmo documento também destaca o entendimento de que esta é uma habilidade necessária para o Século XXI, em razão de seu impacto, advindo dos avanços tecnológicos, nos setores produtivos e postos de trabalho, com a automatização de trabalhos e o surgimento de

demandas por conhecimentos computacionais.

Ainda sobre a relação do aprendizado dos fundamentos de Computação e Pensamento Computacional com outras disciplinas, o documento afirma:

O conhecimento de fundamentos computacionais e o desenvolvimento do pensamento computacional envolvem não somente a capacidade de construir modelos abstratos (de informação e processos) e de sistematizar a solução de problemas, mas também as habilidades de argumentação, análise crítica e trabalho cooperativo. É que o desenvolvimento desses saberes opera na encruzilhada dos conhecimentos de outras disciplinas.

De modo semelhante, Resnick (2013) vê a programação como uma extensão da escrita. Para ele, a habilidade de codificação permite “escrever” novas produções, como histórias interativas, jogos e simulações. E, assim como quem aprende a escrever aprende também a organizar, refinar e refletir suas ideias, quem aprende a programar não só está aprendendo a codificar, está codificando para aprender, isto é, está igualmente aprendendo estratégias de solução de problemas, criação de projetos e comunicação de ideias. Habilidades úteis não apenas para cientistas da computação, mas para todos, independente de idade, procedência, ocupação ou interesse.

Em *International Society for Technology in Education e Computer Science Teachers Association (2011)* o Pensamento Computacional é descrito como uma habilidade essencial para todos os alunos, que envolve a expressão de soluções como uma série de etapas para automatizar um processo e consiste em quatro pilares: decomposição de problemas, reconhecimento de padrões, abstração e algoritmos.

Em seu trabalho, Brackmann (2017) percebe o uso destes quatro pilares pelo Pensamento Computacional para a resolução de problemas. A decomposição diz respeito à maneira que um problema complexo pode ser subdividido em problemas menores e mais facilmente tratados.

Cada um destes problemas podem ser analisados para o reconhecimento de problemas anteriormente solucionados, através do reconhecimento de padrões. Por meio da abstração, informações importantes podem ser destacadas, enquanto detalhes irrelevantes para a solução são deixados em segundo plano.

Assim, com a solução dos problemas menores e mais simples, regras e passos para a solução podem ser encontrados, podem ser criados algoritmos que podem ser entendidos por computadores e utilizados para a solução de problemas complexos.

Por se tratar de uma habilidade que pode ser aplicada em diversas áreas do conhecimento, o Pensamento Computacional não deve estar delimitado ao Ensino Superior, e deve ser estudado desde o Ensino Básico, conforme definido pelas Diretrizes para o ensino de Computação na Educação Básica (Ribeiro *et al.*, 2019).

### **2.3 Diretrizes para ensino de Computação na Educação Básica**

As Diretrizes da Sociedade Brasileira de Computação para o Ensino de Computação na Educação Básica (Ribeiro *et al.*, 2019) defendem a Computação como disciplina essencial na Educação Básica, destacando seu papel na formação de cidadãos críticos, criativos e preparados para os desafios do século XXI. Este documento apresenta a Computação como uma ciência, porque possui fundamentos e princípios que organizam de forma sistemática o conhecimento humano.

Estas diretrizes estabelecem a Computação como uma ciência tanto natural quanto artificial. Natural porque existe muito antes da invenção dos computadores, como nos casos da codificação e processamento do DNA, nos fluxos de informação da Economia e Administração e nas receitas, roteiros e instruções de diversos fins. Artificial porque lida com dimensões abstratas e virtuais, mas que são fundamentais para o mundo real, como processos de informação, investigação de problemas e construção de soluções, tendo como exemplo disso a internet. Assim a Computação não apenas explica o mundo, como também fornece ferramentas para transformá-lo.

Apesar de ser uma área em constante evolução, seus princípios fundamentais são os mesmos há décadas. Dominar estes conceitos fundamentais contribui para um entendimento profundo do mundo, que leva os estudantes a desenvolverem autonomia, flexibilidade, resiliência, proatividade e criatividade. Assim, a Computação se revela não apenas como uma disciplina técnica, mas como um pilar do conhecimento moderno, essencial para formar cidadãos críticos e capazes de intervir no mundo.

O documento também destaca a Computação como uma disciplina transversal pois provê habilidades distintas de outras áreas do conhecimento, e organiza a Computação em três eixos: Pensamento Computacional, Mundo Digital e Cultura Digital.

Pensamento Computacional é definido como a capacidade de compreender, definir, modelar, comparar, solucionar, automatizar e analisar problemas de forma metódica sistemática usando algoritmos, conforme discutido na Seção 2.2.

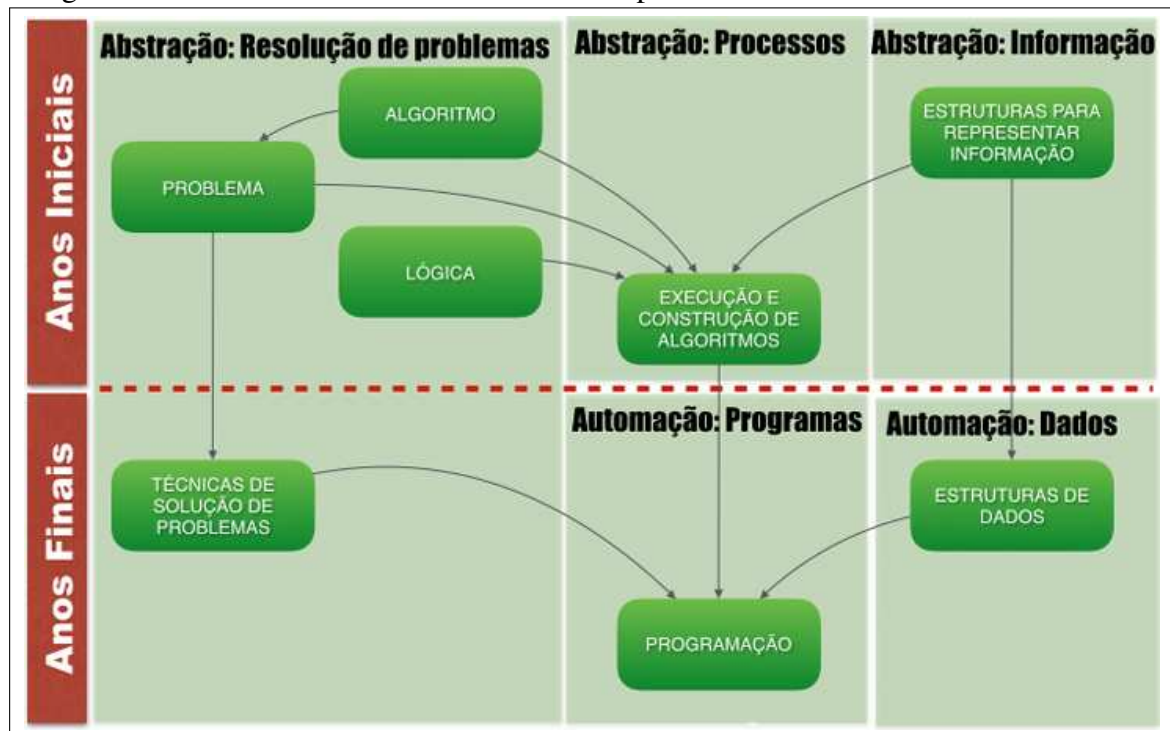
O eixo do Mundo Digital compete ao entendimento do funcionamento e importância da codificação, processamento, distribuição, armazenamento e proteção da informação.

Já a Cultura Digital vai além da simples recepção do conhecimento técnico, mas a reflexão sobre impactos éticos, sociais e econômicos das tecnologias, isto é, um letramento digital, que busca compreender as relações da Computação com outras áreas do conhecimento, promover a fluência no uso do conhecimento computacional para expressão de soluções e manifestações culturais de forma contextualizada e crítica.

Os conteúdos sugeridos pelas diretrizes podem ser ministrados por professores de qualquer área com formação adequada em Computação. Sugere-se que as redes de ensino adaptem os conteúdos à realidade local, integrando-os a disciplinas existentes ou criando uma disciplina específica, tendo total flexibilidade para construir seus currículos da forma que acharem mais adequada.

Os principais conceitos que as diretrizes sugerem ser trabalhados no Ensino Fundamental são agrupados por eixo e estão ilustrados pelas Figuras 1, 2, e 3. Nessas figuras são mostradas as habilidades a serem trabalhadas por temática e por ano de formação dos alunos.

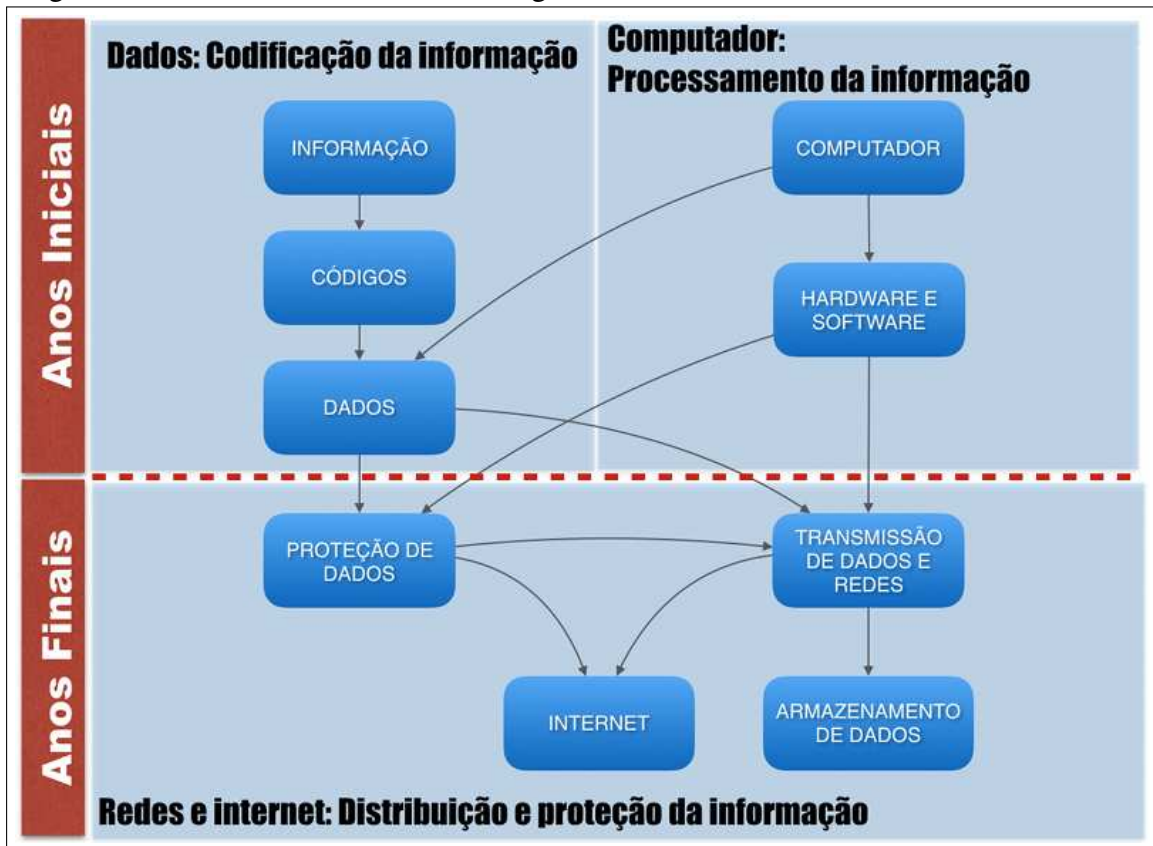
Figura 1 – Conceitos do eixo Pensamento Computacional no Ensino Fundamental



Fonte: Ribeiro *et al.* (2019).

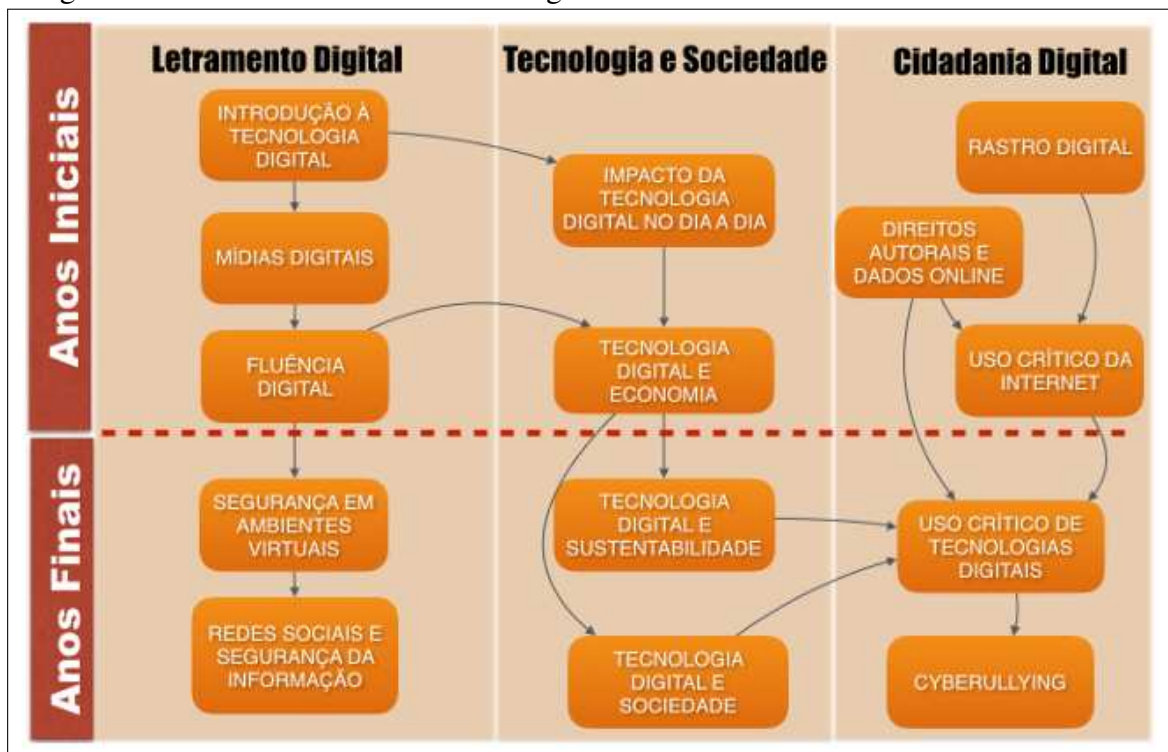
Assim, argumentam as diretrizes, que a Computação contribui para a formação dos jovens, pois permite a compreensão plena do mundo, aumenta a capacidade de aprendizagem e

Figura 2 – Conceitos do eixo Mundo Digital no Ensino Fundamental



Fonte: Ribeiro *et al.* (2019).

Figura 3 – Conceitos do eixo Cultura Digital no Ensino Fundamental



Fonte: Ribeiro *et al.* (2019).

de resolução de problemas, promove a formação de cidadania crítica, prepara para o mercado de trabalho e apoia o aprendizado das demais disciplinas, o que afirma sua importância para a formação plena do cidadão do século XXI.

Anexo às Diretrizes apresentadas, a Base Nacional Comum Curricular (BNCC) Computação (Ministério da Educação, 2022a) define os conteúdos e habilidades relacionados à Educação Digital que devem ser abordados nas escolas e as organiza em tabelas (Anexo A), agrupadas por ano de formação e apresentadas por eixo da Computação, objeto de conhecimento, habilidade, explicação da habilidade e exemplos de como trabalhar uma ou mais habilidades.

Uma ferramenta para suprir a falta de computadores e de *internet* identificada no cenário de aplicação desta pesquisa e para oferecer uma experiência de aprendizado de programação introdutório, demanda as seguintes características:

- a) “Ser totalmente gratuito”;
- b) “Funcionar *offline*”;
- c) “Estar traduzido para português”;
- d) “Ser de fácil aprendizado”, com comandos sucintos e explicados;
- e) “Estar disponível para Android e IOS”.

Dois ferramentas identificadas que pode ser utilizada para introduzir conceitos de computação, inclusive o Pensamento Computacional, junto a alunos do Ensino Fundamental, seguindo os critérios estabelecidas, foram o OctoStudio (Lifelong Kindergarten, 2025) e o ScratchJr (ScratchJr, 2014).

Apesar de ambas as plataformas atenderem os requisitos, o OctoStudio oferece uma maior gama de comandos e uma maior variedade de interações com o dispositivo móvel. Enquanto o ScratchJr oferece o uso da câmera e do microfone do *smartphone*, o OctoStudio oferece funções de vibrar o *smartphone*, acompanhar a inclinação do aparelho, interagir com ímãs, acessar o bluetooth, dentre outras funções.

## 2.4 OctoStudio

O OctoStudio (Lifelong Kindergarten, 2025) é uma ferramenta gratuita desenvolvida pelo grupo de pesquisa *Lifelong Kindergarten* (LLK) do *Massachusetts Institute of Technology* (MIT) *Media Lab* que consiste em um ambiente de programação visual que permite a criação de projetos animados e interativos utilizando sensores dos dispositivos móveis. Seu método de programação ocorre por meio de blocos de comando que são encaixados uns nos outros,

formando uma sequência de instruções, ou seja, um algoritmo (Figura 4).

Figura 4 – Exemplo de algoritmo implementado com blocos no OctoStudio.



Fonte: Biblioteca de projetos do OctoStudio (Massachusetts Institute of Technology, 2025).

O aplicativo já foi traduzido para mais de 30 idiomas e foi desenvolvido com o propósito de democratizar o acesso à programação. Além de ser otimizado para funcionar mesmo em *smartphones* mais antigos e funcionar de forma offline, o que amplia seu alcance e viabiliza o desenvolvimento de projetos por pessoas com acesso limitado a computadores e à internet.

Os trabalhos de Xisto (2023) e Rusk *et al.* (2024), afirmam que o estímulo para a criação deste ambiente de codificação para dispositivos móveis surgiu durante a pandemia de COVID-19 em 2020, quando educadores e parceiros do LLK de países como Brasil, Índia, México e África do Sul compartilharam com o grupo seus desafios para continuar suas atividades

de programação quando a maioria dos alunos não tinha acesso a computadores.

Eles compartilharam suas preocupações sobre a crescente discrepância de oportunidades de aprendizado para as crianças, notando que as crianças gastavam cada vez mais tempo nos celulares assistindo a vídeos, mas faltavam ferramentas projetadas para envolvê-las ativamente na criação de projetos para expressar suas ideias e desenvolver suas habilidades criativas de resolução de problemas.

Conforme ilustra a Figura 5, a interface do OctoStudio é dividida em quatro áreas: o palco, a área de código, o menu de atores e a paleta de blocos. Na parte inferior da tela, na paleta de blocos, é onde ficam os blocos de comando, agrupados em sete categorias conforme sua funcionalidade. E estes blocos podem ser arrastados e unidos para formar comandos mais complexos na área de código do ator que estiver selecionado no menu de atores. E estes comandos serão executados pelo ator e exibidos no palco.

Rusk *et al.* (2024) destacam que, apesar da semelhança com outros ambientes de codificação baseada em blocos, como o Scratch (SCRATCH, 2023) (popular aplicação criada pelo mesmo grupo de pesquisa) o OctoStudio foi projetado para facilitar aos iniciantes e para permitir que aprendam com a prática, além de aproveitar as funcionalidades e os sensores dos dispositivos móveis, ampliando a possibilidade de criação e aprendizado.

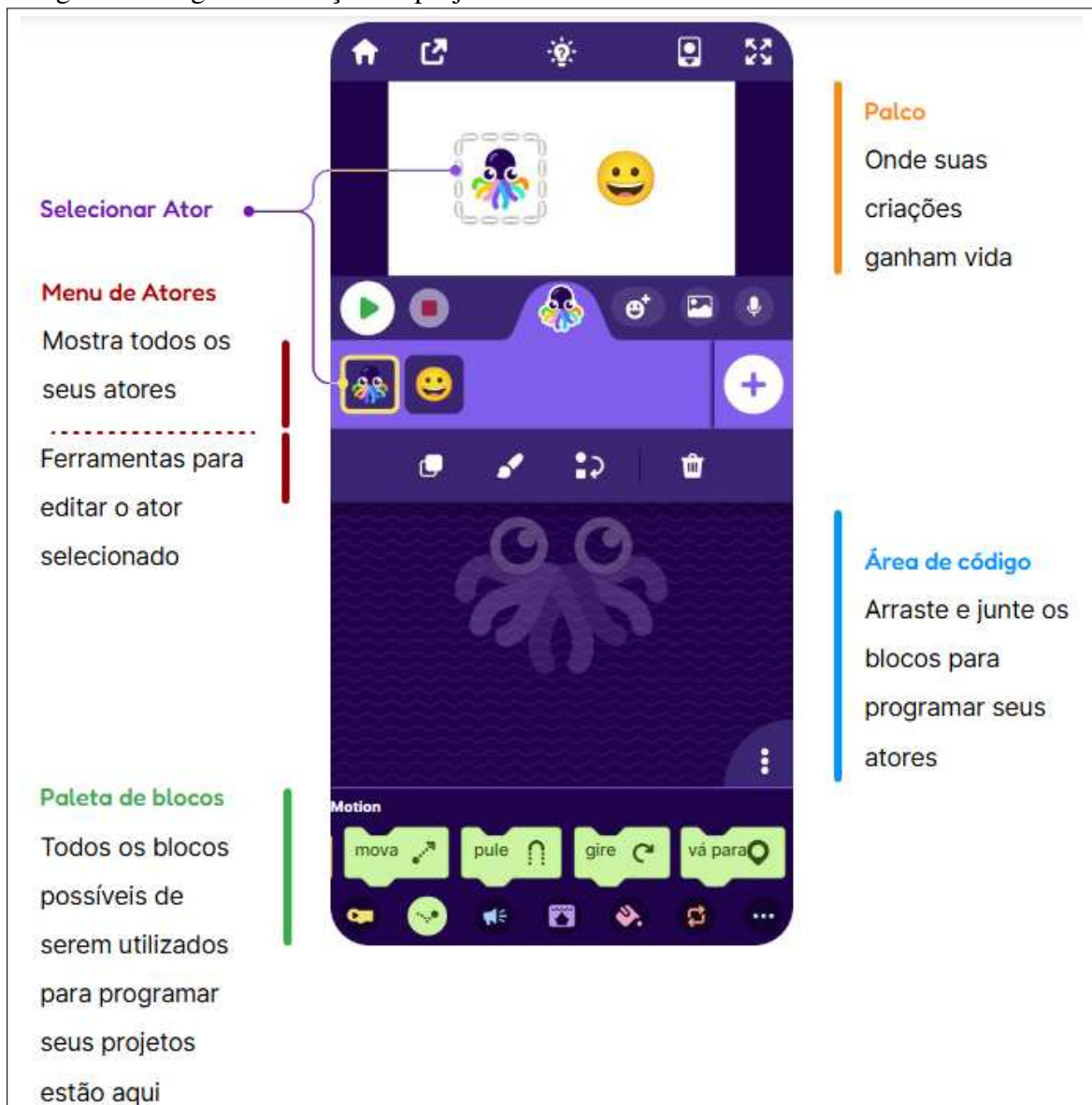
O acesso aos sensores do dispositivo móvel permite que os usuários interajam com o mundo físico, representando-o em seus projetos ou integrando seus projetos a ele, criando possibilidades para programas que funcionam quando o usuário, por exemplo, sacudir, pular, inclinar, comunicação entre projetos via conexão bluetooth e aproximar um ímã do dispositivo, este último, ativando um gatilho de evento disponível nos comandos do aplicativo.

A aplicação estimula a criatividade permitindo que os usuários importem fotos e sons para seus projetos, além de poderem editar personagens e cenários e lhes dar ações a serem realizadas por meios de instruções de comandos, condições e eventos que permitem apreender a lógica de programação, mesmo para os que nunca tiveram contato com linguagens de programação.

Para Nadal (2024), o aplicativo aprimora a alfabetização digital porque, no processo de criação, as crianças desenvolvem habilidades como resolução de problemas, pensamento computacional e criativo, e aprendem e aplicam conhecimentos computacionais e de programação em um contexto significativo e motivador.

Assim, o OctoStudio se apresenta como uma ferramenta com grande potencial para

Figura 5 – Página de criação de projetos do OctoStudio



Fonte: Guia de Referência do OctoStudio, octostudio.org (2024b).

o ensino de programação e para a promoção do ensino de Pensamento Computacional, ao oferecer uma maneira acessível para que diversas pessoas tenham acesso a um ambiente fácil de programação por meio de dispositivos móveis.

Uma forma de se avaliar a aceitação de uma ferramenta digital no ensino é utilizar o modelo de avaliação chamado dTECT.

## 2.5 Modelo de avaliação dTECT

O modelo de avaliação dTECT (Evaluating TEaching CompuTing), foi desenvolvido por Wangenheim *et al.* (2017) com o objetivo de suprir a falta de modelos de avaliação ou

instrumentos de medição para avaliar a qualidade das iniciativas para o ensino de computação na Educação Básica.

O dTECT é um modelo para avaliação da qualidade de unidades instrucionais focadas no ensino de computação para a Educação Básica com base na percepção dos alunos. Seu objetivo é analisar unidades instrucionais para avaliar a percepção da sua qualidade, a experiência de computação e a percepção da aprendizagem do ponto de vista dos alunos.

Para Wangenheim *et al.* (2017), a qualidade de uma unidade instrucional é atingida quando se realiza seus objetivos de aprendizagem, se promove atividades prazerosas que facilitam a aprendizagem e que desperta uma percepção positiva e interessante referente à computação.

Os autores derivaram, a partir dos objetivos do modelo, perguntas de análise da percepção da qualidade de um curso e como fazer a medição, operacionalizada por um questionário com 13 perguntas (Anexo B) a ser respondido pelos alunos, após o término da unidade instrucional. Para os autores, a avaliação pós-curso oferece a vantagem de ser executada com pouco esforço e de uma forma não intrusiva ao final da unidade instrucional.

Após a análise da validade do modelo, com dados de 16 estudos de caso em 13 diferentes instituições de ensino e respostas de 477 alunos, foram obtidas evidências suficientes para considerar o modelo dTECT confiável e válido para a avaliação de unidades instrucionais para o ensino de computação na Educação Básica, bem como seus itens de medição foram considerados consistentes e precisos para avaliar a qualidade de unidades instrucionais.

## **2.6 Síntese do capítulo**

Esta seção apresentou os conceitos de Pensamento Computacional, sua definição, influência no aprendizado e sua importância. Também foram expostas as Diretrizes da Sociedade Brasileira de Computação para o Ensino de Computação na Educação Básica, a Computação como ciência e disciplina a ser desenvolvida na Educação Básica e sua organização por eixos.

Ademais, foi descrito o aplicativo OctoStudio — ferramenta central para esta pesquisa — que será usado durante o curso para ensino dos objetos de conhecimento “Linguagem visual de programação” e “Técnicas de solução de problemas: decomposição”, definidas no eixo de Pensamento Computacional contido na BNCC Computação (Ministério da Educação, 2022a).

Por fim, foi mostrado o modelo de avaliação de unidade instrucional dTECT, que será aplicado com os alunos ao fim do curso proposto nesta pesquisa como forma de validar a qualidade do OctoStudio como ferramenta para o ensino de Pensamento Computacional.

### **3 REVISÃO BIBLIOGRÁFICA**

#### **3.1 Introdução**

Esta seção apresenta a revisão literatura de práticas relacionadas ao tema deste trabalho. A metodologia da pesquisa bibliográfica é descrita na Seção 3.2, os resumos dos trabalhos selecionados são apresentados na Seção 3.3 e as considerações finais da pesquisa são expostas na Seção 3.4.

#### **3.2 Metodologia da pesquisa**

Com o objetivo de identificar os principais trabalhos relacionados a esta pesquisa e alcançar um referencial científico seguro e confiável, foi necessária a realização de uma Revisão Sistemática da Literatura, que foi baseada nos passos definidos por Shimasaki e Prado (2021) e está organizada nas seguinte etapas: identificação da problemática (Seção 3.2.1), escolha do repositório (Seção 3.2.2), estabelecimento de critérios de inclusão/exclusão dos trabalhos (Seção 3.2.3), caracterização dos trabalhos (Seção 3.2.4), análise dos trabalhos e discutir os dados (Seção 3.2.5), resultado da análise (Seção 3.2.6) e apresentação conclusões (Seção 3.2.7).

##### ***3.2.1 Identificação da problemática***

Atualmente, existem diversas iniciativas voltadas para o ensino de Pensamento Computacional, habilidade destacada por Resnick (2013) como útil para pessoas de qualquer idade, precedência, interesse ou ocupação. Ferramentas, como Scratch, têm apresentado bons resultados no ensino desta habilidade (PINHO, 2024), porém, a falta de acesso a computadores foi identificada como barreira para o desenvolvimento desta habilidade como mostram os trabalhos de Pinho (2024) e Xisto (2023).

O aplicativo OctoStudio busca, como um de seus propósitos, solucionar estas lacunas como uma ferramenta projetada para oferecer um ambiente de aprendizado de programação responsivo para dispositivos móveis (Rusk *et al.*, 2024). Assim, esta pesquisa tem o propósito de identificar os métodos apresentados em trabalhos da literatura, que foram aplicados em práticas de ensino com o OctoStudio e que utilizam o Pensamento Computacional atrelado à outras disciplinas e habilidades.

### **3.2.2 Escolha do repositório**

Para esta revisão, foi escolhido como repositório para pesquisa a plataforma Google Acadêmica (GOOGLE, 2004). Também foi realizada uma busca pelo termo OctoStudio na SBC OpenLib (SOL) (SOL, 2018), a biblioteca digital mantida pela Sociedade Brasileira de Computação (SBC).

### **3.2.3 Estabelecimento de critérios de inclusão/exclusão**

Foram estabelecidos os seguintes critérios para inclusão de trabalho a serem analisados:

1. Trabalhos que contenham, em qualquer campo, o termo “OctoStudio”;
2. Trabalhos de acesso gratuito;
3. Excluídas as pesquisas que não relataram a realização de atividade pedagógica com o OctoStudio.

### **3.2.4 Caracterização dos trabalhos**

Como resultado da busca no Google Acadêmico, foram apresentados 46 trabalhos, enquanto que a busca na SOL resultou em um trabalho que, entretanto, já estava listado nos resultados do Google Acadêmico.

Uma vez feita a seleção segundo os critérios estabelecidos, foram selecionados três trabalhos que apresentam práticas de ensino realizadas com o OctoStudio, incluindo a metodologia abordada e os métodos de avaliação.

Além dos trabalhos selecionados da busca, também foi incluído o trabalho de Pinho (2024) por contribuir com uma metodologia de curso, um método de avaliação do ensino e aprendizagem e conteúdo teórico relevante para essa pesquisa. Este trabalho, inclusive, é uma proposta de trabalho futuro do trabalho de Pinho (2024).

### **3.2.5 Análise dos trabalhos e discussão dos dados**

Durante esta etapa foram lidos os trabalhos selecionados, para conhecer suas referências teóricas, contribuições e resultados. Estes trabalhos foram resumidos e estão na Seção 3.3.

### **3.2.6 Resultado da análise**

Ao analisar os trabalhos selecionados e a interface e proposta do OctoStudio, foi mudada a proposta inicial de aplicar este estudo em uma turma de ensino médio, e foi decidido realizar o estudo com alunos do ensino fundamental, como é feito nos trabalhos estudados. Também foi decidido realizar um curso de curta duração como é feito nos trabalhos de Xisto (2023) e Pinho (2024).

### **3.2.7 Apresentação das conclusões**

Após a análise dos trabalhos, foi decidido elaborar uma metodologia voltada para a aplicação do OctoStudio para o ensino fundamental.

Espera-se que este estudo contribua com a identificação do OctoStudio como ferramenta eficaz para o ensino de Pensamento Computacional e apoio ao ensino de disciplinas, além de contribuir para o desenvolvimento do campo de estudo de computação na educação, apresentando questões para pesquisas futuras.

## **3.3 Trabalhos relacionados**

Nesta Seção estão os resumos dos trabalhos após a Revisão Sistemática da Literatura.

### **3.3.1 Estimulando o pensamento computacional no ensino médio utilizando o software Scratch**

Este trabalho investiga a utilização do Scratch como ferramenta para o ensino de programação e desenvolvimento do Pensamento Computacional em alunos do primeiro ano do ensino médio da turma do curso de Desenvolvimento de Sistemas da Escola Estadual de Educação Profissional Professor Walquer Cavalcante Maia, no município de Russas, no interior do Ceará.

O trabalho tem por objetivo explorar o uso do Scratch como ferramenta para ensino de pensamento computacional através da aplicação de aulas de programação, estruturado com desafios progressivos e com a criação de um projeto. Para alcançar este objetivo, Pinho propôs o desenvolvimento de uma metodologia, sua aplicação, sua avaliação e, por fim, a identificação de desafios e oportunidades.

Pinho define o Pensamento Computacional como uma forma de abordar problemas e processos que incorpora conceitos e técnicas fundamentais da ciência da computação. Essa abordagem envolve a capacidade de decompor um problema complexo em partes menores, identificar padrões, desenvolver algoritmos para resolver essas partes e projetar soluções eficientes.

O trabalho utiliza o modelo dTECT (*Evaluating TEaching CompuTing*) desenvolvido por Wangenheim et al. (2017, apud Pinho, 2024), para coletar, por meio de questionários, percepção dos alunos sobre a qualidade em termos de experiência de computação e sua percepção de aprendizagem.

Para a execução do trabalho, Pinho planejou sete aulas teórico-práticas onde foram introduzidos os conceitos de Pensamento Computacional e aos fundamentos da computação utilizando relacionado à ferramenta Scratch e estabelecendo paralelos com a programação convencional, e desafios relacionados ao conteúdo abordado em aula.

Uma vez finalizado o desafio, os alunos apresentavam suas soluções, que eram discutidas em conjunto com a turma, e o tutor apresentava a sua solução para o desafio. Ao fim das aulas, eram propostos desafios com complexidade baseada no progresso da turma com o objetivo de consolidar o aprendizado e estimular o raciocínio lógico. Segundo Pinho, essa abordagem permitiu uma avaliação contínua e forneceu *feedback* imediato aos alunos.

Pinho propõe dois tipos de avaliação, uma destinada à avaliação do aprendizado dos alunos, com as atividades práticas e a dinâmica das aulas finais, e outra voltada para a avaliação do experimento, com um questionário inicial e outro questionário de avaliação.

Para as duas últimas aulas, Pinho aplicou uma dinâmica que consistiu na criação de um programa em Scratch, utilizando os conteúdos aprendidos durante as aulas. Os alunos foram divididos em equipes de dois ou três integrantes e tiveram a liberdade de desenvolver um jogo, uma ferramenta ou uma história utilizando a linguagem de programação Scratch.

Foram oferecidas premiações às três equipes melhor avaliadas por Pinho, um professor da escola e os monitores das aulas. Todos os participantes receberam um *kit* de participação e um certificado de participação do curso.

A avaliação dos programas levou em consideração os seguintes critérios: abrangência dos conteúdos vistos em sala, objetivo claro do projeto, resolução de problemática que evidencia a aplicação prática do Pensamento Computacional e originalidade, não permitindo cópias da internet.

Pinho afirma que, ao participarem das duas modalidades de avaliação, os alunos

consolidaram os conhecimentos desenvolvidos ao longo das aulas. Assim, os resultados prévios das avaliações indicam que o experimento cumpriu seus objetivos, promovendo o aprendizado significativo dos alunos em programação e Pensamento Computacional.

Para avaliar o experimento, foram aplicados dois formulários, disponibilizados aos alunos via Google Forms, um para traçar o perfil socioeconômico e conhecimentos pré-curso dos participantes e outro para coletar a avaliação dos alunos sobre a experiência de aprendizagem, o conteúdo abordado e pontos a serem melhorados.

Pinho analisou os dados coletados na pesquisa buscando identificar as relações entre os conhecimentos adquiridos, o perfil dos alunos e suas percepções sobre programação, e, com base nas análises, o autor obteve informações sobre a eficácia do ensino de programação para promover habilidades de Pensamento Computacional no contexto do ensino médio.

O autor afirma que as aulas ministradas representaram uma experiência educacional enriquecedora para os alunos da turma de Desenvolvimento de Sistemas. Sua pesquisa demonstrou o potencial da tecnologia como uma ferramenta eficaz para o ensino de conceitos complexos de computação, ao mesmo tempo em que incentivou o engajamento dos alunos e promoveu o desenvolvimento de habilidades cruciais de Pensamento Computacional.

Pinho destaca, ainda, como a desigualdade de acesso a recursos tecnológicos, em especial ao computador, pode ser um fator determinante para o desempenho dos alunos gerando disparidades no desempenho dos alunos e, conseqüentemente, limitar ou alavancar as oportunidades de desenvolvimento de habilidades digitais e resultados.

Este trabalho contribui com importantes informações sobre Pensamento Computacional, metodologia de ensino de programação e avaliações sobre o perfil socioeconômico dos alunos e do método do curso. Além de destacar a influência do acesso a recursos tecnológicos no desempenho dos alunos, levantando oportunidades sobre o uso de ferramentas complementares para suprir a falta de acesso a computadores, proporcionando oportunidades de contato com a programação para aqueles que não possuem computadores em casa, promovendo a democratização do acesso a esse conhecimento.

### ***3.3.2 Computação Criativa com OctoStudio no Ensino Fundamental***

Este artigo trata sobre uma atividade de Computação Criativa utilizando o OctoStudio voltada para crianças do 3º ao 5º ano do ensino fundamental, alinhada às diretrizes da BNCC sobre Computação no desenvolvimento de habilidades relacionadas ao Pensamento Computacional.

O objetivo da atividade foi desenvolver o pensamento computacional e a criatividade dos alunos por meio da construção de personagens físicos, a criação de histórias e sua programação no OctoStudio.

Como objetivos específicos são listados: explorar a criação artística e manual com massinha de modelar, estimular a criatividade ao imaginar histórias para os personagens, introduzir conceitos básicos de programação por blocos, digitalizar e animar personagens fictícios no OctoStudio, explorar o uso de sensores e interatividade nos projetos, incentivar a colaboração e a troca de ideias e estimular a experimentação e o aprendizado baseado na exploração e no erro.

Como metodologia, foram seguidas cinco etapas baseadas na Espiral Criativa de Mitchel Resnick, isto é: Imaginar, Criar, Brincar, Compartilhar e Refletir.

Na etapa **Imaginar**, os alunos foram apresentados ao OctoStudio e às suas possibilidades, e foi feito um *brainstorming* conduzido pelo professor auxiliar, incentivando os alunos a pensarem sobre seus personagens e histórias.

Na etapa **Criar**, os alunos criaram seus personagens com massinha de modelar. Os personagens foram fotografados para serem digitalizados no OctoStudio. Os estudantes começaram a programar as animações e interações de seus personagens.

Na etapa **Brincar**, as criações dos alunos foram testadas e suas ideias, personagens e histórias foram ajustadas. Suas soluções também foram comparadas com as dos colegas, promovendo novas ideias e abordagens.

Na etapa **Compartilhar**, cada aluno ou grupo apresentou seu projeto para a turma, explicando a história do personagem, os desafios enfrentados e as soluções encontradas. Nessa etapa, os alunos trocaram ideias e aprenderam uns com os outros, sugerindo melhorias aos projetos dos colegas.

Na etapa **Refletir**, os alunos participaram de uma discussão sobre os desafios enfrentados e os aprendizados adquiridos. Eles refletiram sobre o que funcionou bem e o que poderia ser melhorado. Além disso, os alunos foram incentivados a registrar suas descobertas e ideias para futuras atividades, reforçando o caráter iterativo da Espiral Criativa e o aprendizado contínuo.

Como avaliação, foram feitas observações diretas ao longo de todas as fases da atividade. Durante as três primeiras etapas, os educadores acompanham a participação dos alunos, avaliando sua capacidade de colaboração, criatividade e narrativas, além da aplicação dos conceitos de programação.

Também foi aplicada autoavaliação e reflexão, incentivando os alunos a registrarem seus desafios e aprendizados ao longo do processo. Durante a fase de Compartilhar, foi feita a avaliação por pares e trocas de ideias.

Para Batista *et al.*, os diferentes métodos de avaliação fomentam um ambiente de crescimento contínuo, colaboração e criatividade, o que garante uma experiência educacional enriquecedora e alinhada às diretrizes da Computação na Educação Básica.

Apesar de não apresentar resultados, o trabalho contribui com uma estrutura de metodologia para a prática com o OctoStudio alinhada às diretrizes da BNCC sobre computação e técnicas de avaliação dos alunos e do projeto.

### ***3.3.3 The Combination of Programming Language with Artistic Language as a Teaching Resource to Develop Creativity and Digital Competence***

Este estudo realizou um teste piloto para analisar e avaliar se há melhora nas competências digitais e criativas dos estudantes do quarto ano do ensino fundamental da escola Taialà situada no bairro de Taialà de Girona, através do projeto artístico chamado “*Making Faces*”, um projeto do ilustrador e artista israelense Hanoch Piven, que consiste em criar personagens a partir de objetos cotidianos, aplicado à programação por meio da aplicação OctoStudio.

O trabalho introduziu esse projeto artístico no âmbito digital e, com a aplicação de programação OctoStudio, combinou código e arte para potencializar a criatividade manipulativa e digital e a motivação pela programação.

A pesquisa consiste em fases de investigação nas quais os resultados foram documentados, analisados e interpretados com base em variáveis estudadas, que foram: a avaliação feita pelos estudantes sobre a experiência educacional deles, a autoavaliação do progresso no aprendizado da linguagem de programação e o nível de aquisição de habilidade em programação e criatividade.

O trabalho traz a definição de Pensamento Computacional e programação, sua importância para todos os indivíduos, sua relação com escrita, comunicação em linguagem natural, pensamento criativo e desenvolvimento cognitivo e social das crianças. Além disso, o trabalho também lista estratégias para melhorar a experiência de aprendizagem: aumentar o uso de vocabulário computacional, aceitação de tentativa e erro, trabalho em equipe e grupos de negociação de soluções.

Os alunos da escola Taialà são de diferentes gerações de famílias trabalhadoras de

níveis socioeconômicos médio-baixo que sempre morou no bairro. O projeto de pesquisa foi posto em prática com esses alunos em duas turmas de vinte e cinco alunos, organizadas em quatro grupos de doze e treze alunos para ter espaço suficiente para realizar o projeto planejado.

Antes da realização do projeto proposto, foram feitas duas sessões nas quais fizeram pequenos projetos livres com Scratch Junior, uma aplicação inspirada na linguagem de programação Scratch, desenvolvida pelo *Developmental Technologies (DevTech) Research Group* e projetada de forma simplificada para ser mais apropriada para crianças menos. Ou seja, a partir da experimentação livre dos alunos, eles criaram seus projetos, sem saber exatamente o que estavam programando.

Para a implementação do projeto, foi elaborada uma sequência didática de quatro sessões para que os alunos pudessem conhecer a aplicação e realizar o projeto *Making Faces*. As Figuras 6, 7 e 8 mostram registros das duas últimas sessões do projeto.

Figura 6 – Alunos fazendo as suas criações artísticas



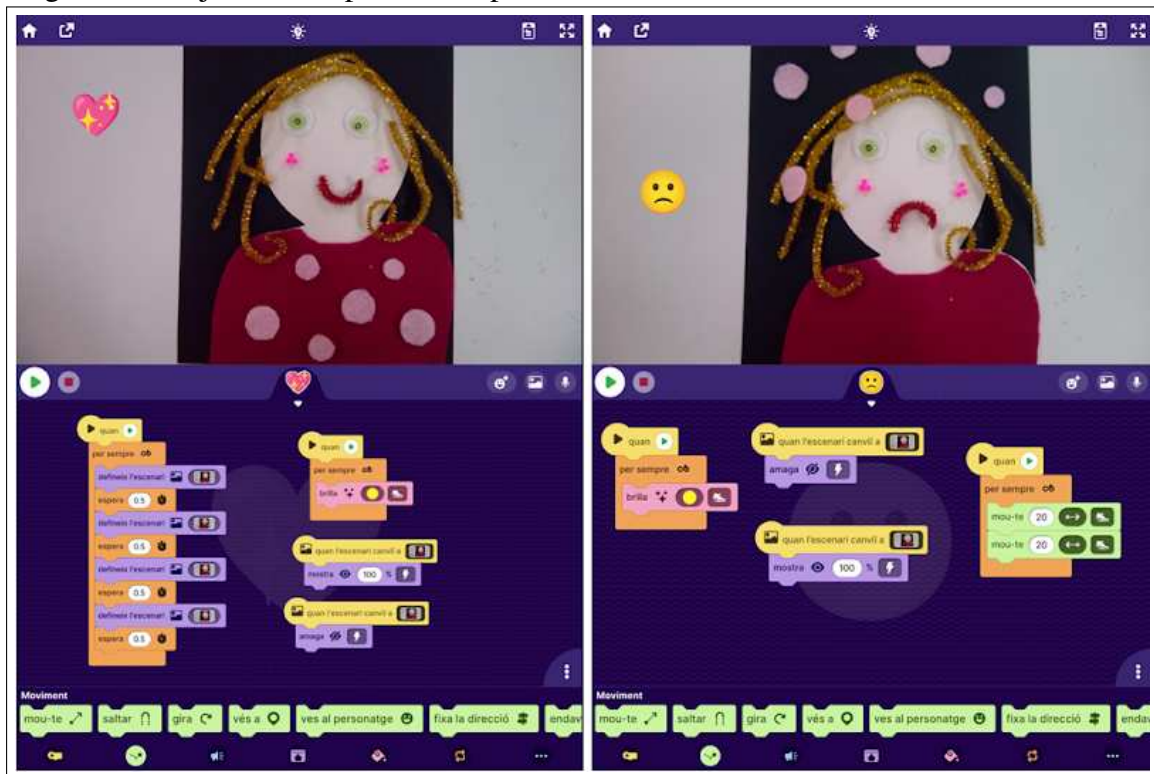
Fonte: Nadal (2024).

Figura 7 – Criações artísticas dos alunos



Fonte: Nadal (2024).

Figura 8 – Projeto criado por uma dupla



Fonte: Nadal (2024).

Durante a primeira sessão os alunos experimentaram livremente o aplicativo OctoStudio, investigando os blocos de programação, quais opções poderiam fazer e criando seus projetos.

Na segunda sessão, eles tiveram que resolver pequenos desafios para aprender a usar todas as categorias de blocos de programação do OctoStudio, fomentando sua motivação para programar e aprender.

Na terceira sessão, eles assistiram a um pequeno vídeo de Hanoch Piven sobre o *Making Faces* mostrando o processo que ele seguiu para criar um rosto, através do método de tentativa e erro, o projeto foi explicado a eles e um exemplo de amostra foi apresentado para servir de referência. Depois, eles começaram a fazer suas criações em duplas com os materiais que já haviam trazido de casa e tiraram diferentes fotografias com a técnica de animação *stop-motion*, movendo elementos do rosto para criar a animação da semana seguinte.

Por fim, na última sessão, eles programaram seu projeto para atingir o desafio de criar uma animação para simular o movimento do rosto e criar realismo.

Para a coleta de dados, foram utilizadas quatro técnicas de coleta: observação direta, que coletou ações e reações dos alunos durante as sessões programadas dentro do projeto, notas de campo, que registrou comportamentos, comentários e expressões dos alunos, e duas rubricas

de avaliação, uma para avaliar se os alunos entenderam o que fizeram e avaliar o resultado, e outra para avaliar o processo de execução e o trabalho em equipe.

Os alunos realizaram uma autoavaliação que os permitiu ver e analisar seu projeto, sua experiência e sua aprendizagem, ou seja, que nota dão a tudo o que aprenderam ao longo do projeto.

Ao analisar resultados obtidos, Nadal conclui que o projeto fomentou a criatividade e o uso da linguagem computacional, considerou que o OctoStudio é uma boa ferramenta para motivar os alunos a continuar aprendendo sobre programação, e que o projeto “*Making Faces*” aplicado ao campo digital através da aplicação de programação OctoStudio teve um impacto positivo na desenvolvimento da competência digital, já que aprenderam a utilizar a linguagem de programação para realizar o projeto e dar soluções aos desafios propostos, e no estímulo à criatividade, já que os alunos tiveram que olhar os objetos cotidianos que carregavam sob uma nova perspectiva e dar-lhes um novo uso para construir os rostos.

O trabalho demonstrou a eficiência do OctoStudio como ferramenta de ensino de programação e estímulo à criatividade, abrindo oportunidades para a aplicação do OctoStudio seja focado no ensino de programação e pensamento computacional ou integrado à prática de outras habilidades e ao ensino de outras disciplinas.

### **3.3.4 *Playful Occupations: Mobile Creative Coding for Critical Consciousness***

A tese investiga como a codificação criativa pode apoiar o desenvolvimento da consciência crítica e promover a justiça social em comunidades do Movimento dos Trabalhadores Sem Teto (MTST), um movimento social populista de esquerda no Brasil, cujo o foco principal é combater a desigualdade habitacional por meio de ocupações massivas e coordenadas de terras privadas improdutivas.

Como base teórica de seu trabalho, Xisto traz os princípios da pedagogia libertadora de Paulo Freire ao contexto moderno com as ideias de Andrea diDessa sobre alfabetização computacional e a teoria do construtivismo de Seymour Papert.

Xisto trabalhou com o Núcleo de tecnologia do MTST que, segundo a autora, foi pioneiro como movimento de programação a se concentrar em ambientes de ensino e desenvolvimento baseados em celulares, pois a maior parte da base do movimento não tem acesso a computadores, mas muitos têm acesso a smartphones.

O trabalho apresenta o OctoStudio como ferramenta para suprir as os desafios

encontrados pelo Núcleo de tecnologia para escolha de uma plataforma adequada, como a incompatibilidade para telas de dispositivos móveis, a falta de tradução para português, os preços altos e a indisponibilidade para dispositivos antigos ou básicos.

As principais contribuições do trabalho foram a implementação de oito oficinas em quatro ocupações na periferia de São Paulo, juntamente com um evento de celebração final na sede do MTST, e a elaboração do guia Ocupando a Programação: Guia de Projetos no OctoStudio (POP) para suprir demandas de apoio pedagógicos identificadas por Xisto em entrevistas pós-oficina.

Ao longo do trabalho a autora apresenta os artefatos usados para guiar suas oficinas e os elaborados para o guia POP como: os cartões de início de projeto (Figura 9) com sugestões de ponto de partida para implementação, uma atividade quebra-gelo em que os participantes das oficinas se revezam encadeando blocos de comando impressos para um amigo representar, uma sugestão de estrutura de oficina e guia de referência de atividades semanais com o OctoStudio.

Figura 9 – Amostras dos cartões de início do projeto



Fonte: Xisto (2023).

Como conclusão, Xisto destaca os aprendizados que adquiriu com seu projeto para a implementação eficaz de um ambiente criativo de codificação: envolva-se profunda, autêntica e criticamente com o contexto do projeto, crie conteúdo que seja especificamente relevante para a comunidade e cultive um abordagem de aluno como agente e o empoderamento comunitário.

Este trabalho mostra-se importante para o estudo ao apresentar o potencial OctoStudio como ferramenta de ensino e trazer diversas questões que envolvem um projeto de ensino de programação, referindo-se a desafios técnicos, pedagógicos, infraestruturais e sociais enfrentados durante o planejamento e execução do projeto.

### 3.3.5 Análise comparativa

A Tabela 1 apresenta uma síntese das contribuições metodológicas de cada trabalho relacionado, comparados ao método proposto nesta pesquisa, permitindo uma melhor visualização das contribuições deste trabalho.

Tabela 1 – Comparação entre este trabalho e os trabalhos relacionados

<b>Trabalho</b>	<b>Aborda OctoStudio</b>	<b>É voltado para o ensino fundamental</b>	<b>Aborda Pensamento Computacional</b>	<b>Analisa questões Socio-econômicas</b>	<b>Faz paralelo com a disciplina de matemática</b>
Batista <i>et al.</i> (2025)	X	X	X		
Nadal (2024)	X	X	X	X	
Xisto (2023)	X		X		
Pinho (2024)		X	X	X	
<b>Este trabalho</b>	X	X	X	X	X

Fonte: Elaborado pelo autor (2025).

### 3.4 Considerações finais

Os trabalhos apresentados nesta seção mostram-se relevantes para o entendimento do tema desta pesquisa, por suas contribuições teóricas, métodos e resultados apresentados, evidenciando o potencial de contribuição do OctoStudio para suprir demandas também levantadas por estes trabalhos como necessidades dos estudantes.

Percebe-se, portanto, que a abordagem atual de ensino de Pensamento Computacional pode ser enriquecida, as carências infraestruturais domiciliares e escolares podem ser mitigadas e o entendimento sobre o ensino alinhado à computação pode ser ampliado pela contribuição de ferramentas com o OctoStudio.

## **4 METODOLOGIA**

### **4.1 Introdução**

Considerando as problemáticas e objetivos apresentados, esta pesquisa pode ser definida como qualitativa de natureza aplicada, sendo ainda uma pesquisa ação, pois busca investigar a forma como a utilização do OctoStudio em aulas de programação impactam nos processos de ensino e aprendizado de Pensamento Computacional alinhado ao ensino de Matemática para alunos do sexto ano do Ensino Fundamental da Escola Municipal de Ensino Fundamental Evaldo Holanda Maia, da cidade de Limoeiro do Norte, no interior do Ceará.

As impressões apresentadas acima subsidiam-se de acordo com Gerhardt e Silveira (2009), as quais apresentam a pesquisa qualitativa como aquela que busca compreender os aspectos subjetivos do âmbito estudado, atentando-se às formas como o fenômeno se manifesta, enquanto a pesquisa de natureza aplicada visa “gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos” (Gerhardt; Silveira, 2009) - ambas as definições alinham-se às propostas escolhidas. Classifica-se também como pesquisa ação pois pressupõe uma transformação na realidade onde foi realizada a pesquisa (Fonseca, 2002).

Nesta seção, são apresentados, detalhadamente, os procedimentos elencados para o desenvolvimento da pesquisa, com vias a garantir a confiabilidade e a sua reprodutibilidade. Na Seção 4.2, estão descritos os procedimentos adotados com relação aos aspectos éticos da pesquisa. Na Seção 4.3, está a descrição de como se deram as aulas, sua duração prevista, os materiais utilizados, as metodologias das aulas e como foi feita a avaliação do experimento. Na Seção 4.4, está a descrição sobre os questionários aplicados na pesquisa, sua quantidade, natureza e forma de aplicação.

### **4.2 Aspectos éticos da pesquisa**

Visando a garantia da integridade dos colaboradores da pesquisa, foram considerados aspectos éticos legais, os quais se constituem do Termo de Consentimento Livre e Esclarecido (TCLE) (Apêndice A), assinado pelos responsáveis pelos alunos, autorizando sua participação, e do Termo de Assentimento Livre e Esclarecido (TALE) (Apêndice B), assinado pelos alunos, atestando sua voluntariedade na participação da pesquisa. Antes do início das aulas, a proposta da pesquisa foi apresentada aos participantes e a seus responsáveis, visando esclarecer seus

papéis neste processo, destacando os objetivos e metodologia elencados.

Apesar do consentimento dos responsáveis e desejo de participação dos alunos, nesta pesquisa, será utilizado um sistema de nomes padrão de "Aluno 01, Aluno 02, etc.", para resguardar a privacidade dos participantes, em uma ordem diferente da ordem alfabética dos nomes dos alunos. Além disso, todas as fotos e imagens das aulas e atividades que forem incluídas no trabalho terão os rostos dos alunos borrados, garantindo a proteção da identidade dos participantes.

### 4.3 Ementa e desenvolvimento das aulas

Esta seção apresenta uma descrição do plano de desenvolvimento das aulas, o cronograma das aulas, o material utilizado, quais tópicos foram abordados, quais as atividades e avaliações foram aplicadas.

#### 4.3.1 Cronograma das aulas

Foram realizadas quatro aulas entre o meses de outubro e novembro de 2025, durante dois horários de aulas, nos dias de quinta-feira. Ao todo, o curso durou 8 horas, e se distribuiu em 4 semanas.

A organização das aulas teve como base o agrupamento de instruções presente nas habilidades propostas pela BNCC Computação (Ministério da Educação, 2022a), englobando estrutura sequencial, de repetição e de seleção. Assim, o curso está organizado em quatro aulas: uma de introdução e para comandos sequenciais, uma para comandos de seleção e revisão do conteúdo já visto, uma para comandos de repetição e revisão e uma última aula de encerramento para apresentação da atividade prática de avaliação.

O Quadro 1 apresenta os conteúdos abordados por aula e o tipo de aula.

Quadro 1 – Cronograma das aulas

<b>Aula</b>	<b>Conteúdo</b>	<b>Tipo</b>
01	Tema: Introdução e comandos sequenciais	Teórica e prática
02	Tema: Revisão e comandos de controle	Teórica e prática
03	Tema: Revisão e comandos de repetição	Teórica e prática
04	Tema: Apresentação dos projetos	Prática

Fonte: Elaborado pelo autor (2025).

### 4.3.2 *Metodologias das aulas*

As aulas do curso foram organizadas em duas partes: uma teórica e uma prática. Durante a parte teóricas, foram explicados os conceitos de programação com o OctoStudio, organizados por aula como tema principal conforme o tipo de instrução: sequencial, seleção e repetição.

Na parte prática da aula, foram realizadas atividades para a aplicação do conteúdo visto na primeira parte, com a implementação de atividades propostas para os alunos. Os temas abordados nas atividades envolvem principalmente os conteúdos vistos na disciplina de matemática de formas geométricas, números decimais e as quatro operações básicas, podendo também envolver temas em geral para a aplicação dos comandos vistos.

Durante a realização das atividades em sala, os alunos foram acompanhados pelo professor para auxiliá-los na resolução dos exercícios e tirar eventuais dúvidas. O professor realizou anotações para avaliação do desempenho dos alunos e dos conteúdos onde os alunos tiveram mais dificuldade, a fim de obter dados sobre o curso ministrado.

Ao fim de cada aula, se a atividade de sala não tivesse sido completada, ela foi proposta como atividade de casa e sua resolução foi conferida na aula seguinte, com a apresentação da solução implementada pelo aluno, bem como a solução do professor, que foi apresentada para toda a turma.

### 4.3.3 *Materiais utilizados*

Para as partes teóricas das aulas, foram preparados slides (Apêndice C) para a exibição do conteúdo das aulas, que foram apresentados por um projetor. Já para as partes práticas, foram preparados materiais impressos com os slides de resumos do conteúdo visto na parte teórica, contendo também dos slides das atividades propostas da aula. Também foi usado o projetor para apresentar uma possível solução da atividade realizada em sala.

Para a realização das atividades, os alunos foram incentivados a usarem seus aparelhos próprios, de forma que houvesse um aparelho por aluno durante as aulas. Também foram fornecidos um *tablet* e um *smartphone* para os alunos que não conseguiram levar os próprios para a aula.

Porém, ainda foi identificada a necessidade de serem formadas equipes de para a prática dos conteúdos pois, em algumas aulas, nem todos os alunos tinham seus aparelhos

individuais.

Embora a Lei N° 15.100/2025 (BRASIL, 2025) proíba o uso de aparelhos eletrônicos portáteis pessoais durante a aula, o recreio ou intervalos entre as aulas, para todas as etapas da educação básica. Salvo as exceções dispostas nesta lei, o uso destes aparelhos é permitido para fins pedagógicos ou didáticos, conforme orientação dos profissionais de educação.

Assim, antes de ser comunicado aos alunos sobre o curso e o uso dos aparelhos em sala de aula, foi primeiramente comunicado ao diretor da escola e a professora da disciplina sobre a necessidade de uso de celular ou *tablet*. Em seguida, a professora comunicou aos pais dos alunos e foi acertado que o uso dos aparelhos ficaria restrito ao período das aulas do curso, dentro da sala de aula da turma. Só então, os termos foram entregues para a assinatura dos alunos e responsáveis.

#### **4.3.4 Avaliação dos conhecimentos**

Para a avaliação dos conhecimentos adquiridos pelos alunos, foi aplicada uma avaliação prática, com base em duas propostas. A primeira foi que os alunos deviam realizar atividades pré-definidas, em equipe ou individualmente. A segunda foi a realização de um problema definido pelos próprios alunos. Esta atividade foi avaliada sob critérios elaborados com base nas habilidades definidas na BNCC Computação (Ministério da Educação, 2022a) trabalhadas no curso, que são:

- (EF06CO02) Elaborar algoritmos que envolvam instruções sequenciais, de repetição e de seleção usando uma linguagem de programação;
- (EF06CO03) Descrever com precisão a solução de um problema, construindo o programa que implementa a solução descrita;
- (EF06CO04) Construir soluções de problemas usando a técnica de decomposição e automatizar tais soluções usando uma linguagem de programação.

Além desta atividade proposta, ao longo do curso, o tutor também realizou anotações de observações que pudessem registrar a demonstração de algum destes aspectos, e descritos na Seção 5.4.

#### **4.4 Desenvolvimento dos questionários de pesquisa**

Para a realização desta pesquisa, foram utilizados dois questionários, um pré-curso e outro pós-curso, ambos baseados no modelo de avaliação dTECT. O primeiro, realizado antes do curso, é composto totalmente por perguntas fechadas, e tem como objetivos contextualizar a proximidade dos participantes com a programação e compreender suas realidades socioeconômicas, possibilitando traçar o perfil dos alunos. O segundo foi aplicado após a conclusão das aulas, para obter as percepções dos alunos sobre o curso e, assim, suas avaliações sobre a qualidade do curso. Ambos foram impressos e entregues aos participantes para serem respondidos na sala de aula.

#### **4.5 Aplicação das aulas**

##### ***4.5.1 Contextualização da escola e da turma***

A Escola Municipal de Ensino Fundamental Evaldo Holanda Maia está localizada na cidade de Limoeiro do Norte, no interior do Estado do Ceará. A escola oferta o ensino regular no sexto ano e integral do sétimo ao nono ano, atendendo às comunidades e bairros da cidade como: Antônio Holanda, Boa Fé, Bom Nome, Brotolândia, Canafístula, Centro, Ilha de Santa Terezinha, Luiz Alves, Marquinho, Pedra Branca, Pitombeira, Quixaba, dentre outros.

A escola possui uma biblioteca, uma quadra, as salas de aula são equipadas com ar-condicionado e os professores têm à sua disposição equipamentos para o auxílio de suas atividades, como impressora e projetor. Porém, a escola não dispõe de um laboratório de informática, nem computadores para o uso dos alunos, mostrando o potencial do uso de smartphones para o ensino de programação e computação.

A turma onde foi realizado o experimento foi a do 6º C, composta por 29 alunos, mas, apenas 28 participaram, pois um dos alunos faltou os dias com aulas do curso.

##### ***4.5.2 Descrição das aulas***

As aulas foram iniciadas no dia 23 de outubro e foram encerradas no dia 18 de novembro, durante as aulas de matemática, por dois horários na quinta-feira, indo das 9:10 às 11:00, com exceção da última aula, que foi realizada na terça-feira, das 7:00 às 8:50.

Durante as aulas, cada aluno pôde usar um *smartphone* ou *tablet*, ou formar equipe

com algum colega, caso não houvesse um aparelho disponível para ele. Todas as aulas contaram com a presença do tutor do curso, que foi acompanhada pela professora de matemática e sua auxiliar, tendo ocorrido apenas uma vez a ausência da professora por motivos de saúde.

A seguir são detalhadas as aulas realizadas, conforme seus respectivos dias.

#### 4.5.2.1 Aula 01 - dia 23 de outubro de 2025

Na primeira aula, foram apresentados aos alunos da turma o tutor, os objetivos do curso, o método de avaliação, o aplicativo e como este seria avaliado.

O primeiro momento foi dedicado à instalação do aplicativo nos *smartphones* dos alunos que ainda não tinham o aplicativo instalado. Também foram levantados os alunos que não tinham aparelho disponível, que receberiam algum aparelho adicional ou seriam organizados em equipe com outro aluno com *smartphone*. Neste dia, apenas dois alunos não estavam com seus aparelhos e puderam ser supridos com os aparelhos levados pelo tutor.

Em seguida, foi apresentado o aplicativo OctoStudio para os alunos seguindo guia de referência do OctoStudio (Anexo C), disponível em seu site oficial. O guia apresenta uma breve introdução sobre o aplicativo, a interface do aplicativo com suas divisões e nomenclaturas, os comandos disponíveis no aplicativo, e a descrição destes comando e as configurações e informações do sistema.

Uma cópia impressa do guia foi entregue aos alunos para que eles pudessem usar como referência ao longo do curso. Então foram realizados os exercícios propostos pelos cartões de programação do OctoStudio (Anexo D, que pode ser encontrado em Lifelong Kindergarten (2025)), disponibilizados no site oficial do projeto, que exploram muitos dos comandos presentes no aplicativo.

Os alunos puderam realizar oito dos nove exercícios presentes no material com a explicação das atividades focada na ordem com que os comandos era executados, isto é, instruções sequenciais. Apenas a última atividade: Emita um Sinal (Figura 10), não foi realizada por falta de tempo até o horário do fim da aula.

Os exercícios foram realizados juntamente com o tutor, que transmitia a tela do seu *smartphone* com o projetor e os alunos acompanhavam a resolução dos exercícios, tiravam dúvidas e faziam observações. Como exercício para casa, foram passados os tópicos de “Explore mais” dos cartões de programação.

Figura 10 – Atividade emita um sinal



Fonte: Cartões de programação do OctoStudio (OCTOSTUDIO.ORG, 2024a).

Os cartões também foram impressos e entregues aos alunos para que eles pudessem consultar e praticar mais dos comandos ao longo do curso. Assim, na primeira aula a turma pôde-se familiarizar com a interface do aplicativo e com o funcionamento dos comandos presentes.

Por fim, foi entregue aos alunos o questionário pré-curso para que eles pudessem preencher. Aqueles que não conseguiram preencher a tempo do fim da aula, puderam levar o questionário para casa e entregar na aula seguinte.

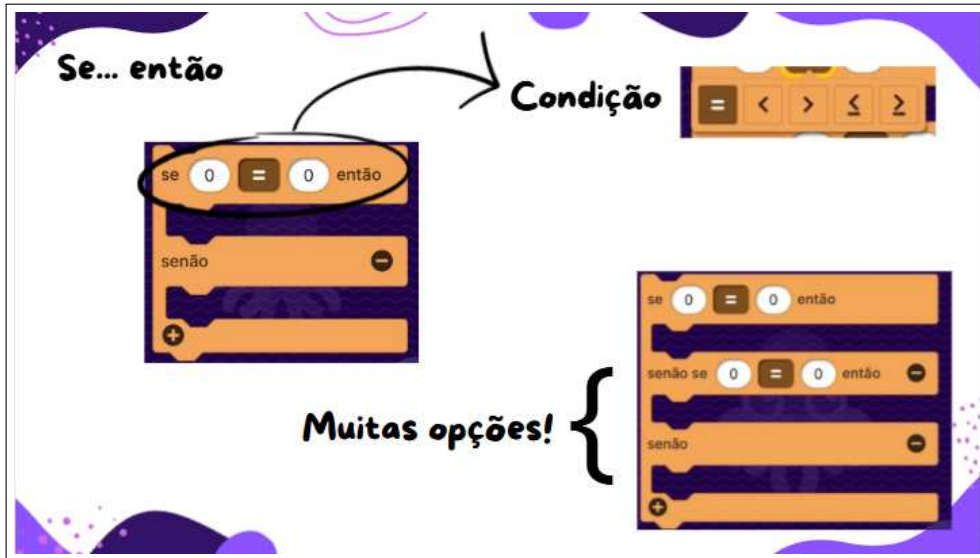
#### 4.5.2.2 Aula 02 - dia 30 de outubro de 2025

Na segunda aula, foram recebidos os questionários e verificados os exercícios dos tópicos de “Explore mais” dos cartões de programação. Nesta aula, três dos alunos não haviam trazido aparelho, assim, dois deles usaram os aparelhos emprestados pelo tutor e outro formou dupla com um deles.

Foram registrados nas anotações do tutor os alunos que conseguiram realizar algum dos exercícios propostos ou alguma atividade própria, demonstrando, assim, aspectos de avaliação apresentados na Seção 4.3.4.

Os conteúdos abordados nesta aula foram uma revisão de alguns comandos já vistos que seriam utilizados nesta aula e comandos de controle, em especial o comando “Se-então-senão”, como o da Figura 11.

Figura 11 – Comando Se-então-senão no OctoStudio



Fonte: Elaborado pelo autor.

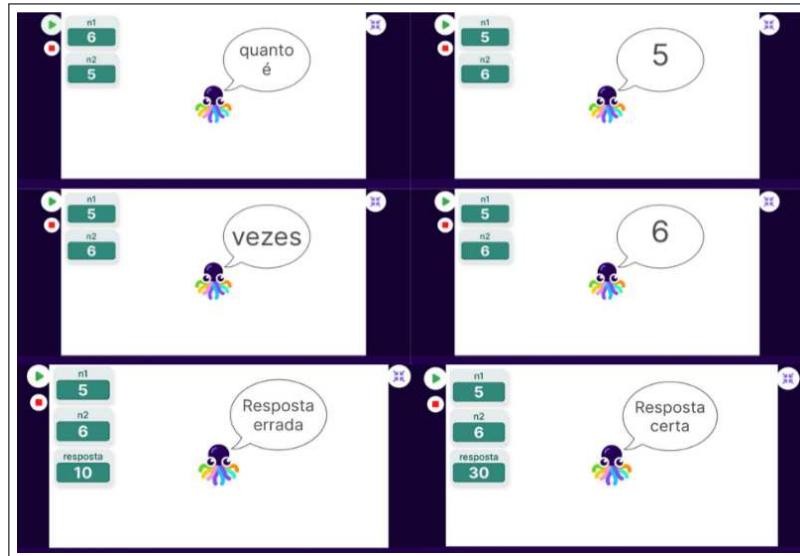
Uma vez apresentado do conteúdo teórico da aula, foram propostos dois exercícios de prática, que foram realizados em conjunto com o tutor, que transmitiu a tela do seu *smartphone* e seguia a elaboração dos programas guiada pelo tutor, mas seguindo as sugestões dadas pelos alunos.

Primeiro, uma atividade que consistia em um programa que multiplicasse dois valores aleatórios e esperasse pela resposta do usuário e, após o tempo de espera, mostrasse uma mensagem se o valor fornecido estava correto ou não. Como mostrado na Figura 12.

Em seguida, como segundo exercício, foi proposto um programa com as quatro operações básicas da matemática, no qual um simbolo representando cada uma sairia da lateral do palco, iria ao centro do palco e exibiria uma mensagem com a resposta de sua operação correspondente entre dois números aleatórios. Esse exercício pode ser visto na Figura 13.

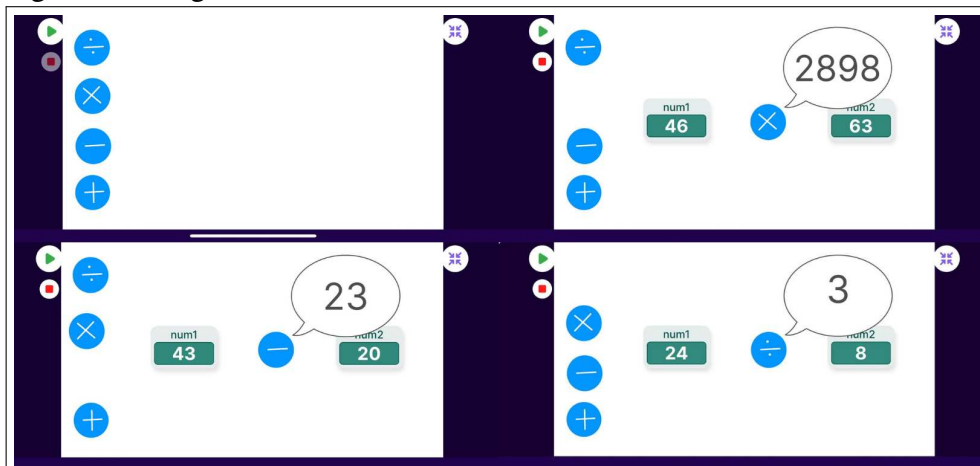
Foi feita em sala de aula a operação de soma e foram passadas como atividade de casa as operações restantes. Durante as práticas dos exercícios, foram feitas anotações conforme os alunos tiravam dúvidas e apresentavam seus programas, para registrar a demonstração alguma das competências listadas na Seção 4.3.4.

Figura 12 – Primeiro exercício da aula de comandos de controle



Fonte: Elaborado pelo autor.

Figura 13 – Segundo exercício da aula de comandos de controle



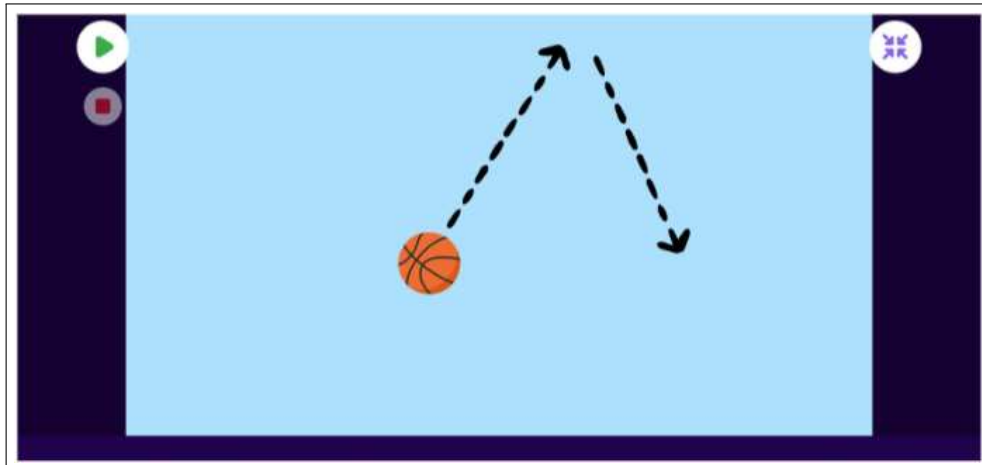
Fonte: Elaborado pelo autor.

#### 4.5.2.3 Aula 03 - dia 06 de novembro de 2025

Nesta aula, foi verificada a realização da atividade da aula anterior e apresenta a solução aos alunos, que puderam acompanhar e realizar em seus próprios aparelhos. Neste dia, apenas um aluno não levou aparelho próprio, pois seu *smartphone* estava com defeito, por isso ele acompanhou a aula com um aparelho emprestado pelo tutor.

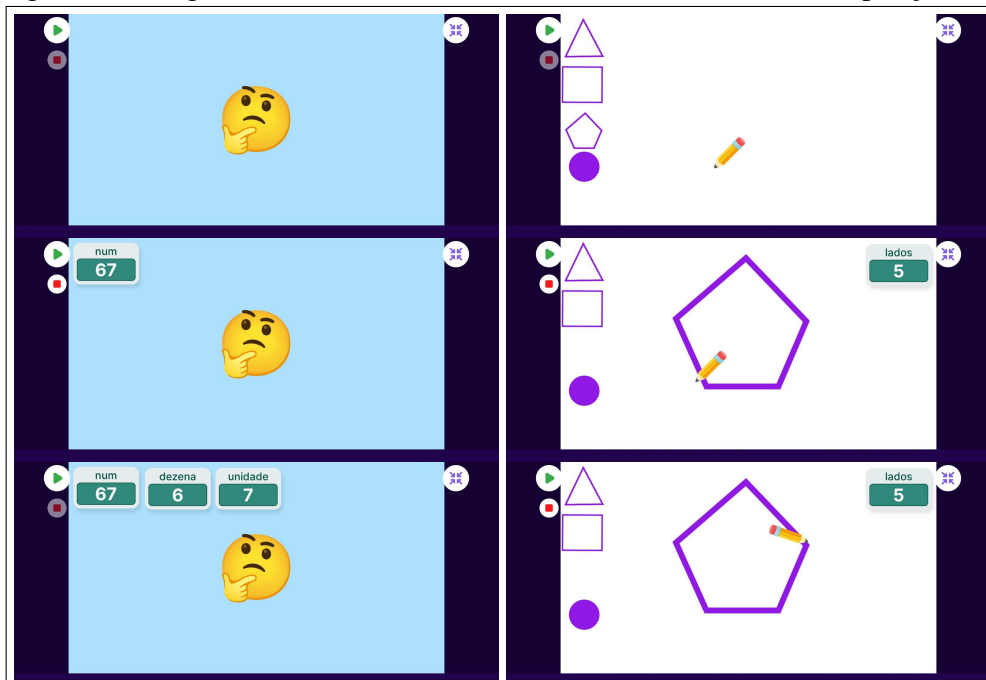
Como conteúdo abordado na aula, foram revisados comandos que seriam usados nas práticas do dia e foram apresentado, de forma mais focada, os comandos de repetição do OctoStudio. Como atividades para fixação do conteúdo, foram realizados exercícios como os mostrados nas Figuras 14, 15a e 15b.

Figura 14 – Primeiro exercício da aula de comandos de repetição



Fonte: Elaborado pelo autor.

Figura 15 – Segundo e terceiro exercício da aula de comandos de repetição



(a) Segundo exercício

(b) Terceiro exercício

Fonte: Elaborado pelo autor.

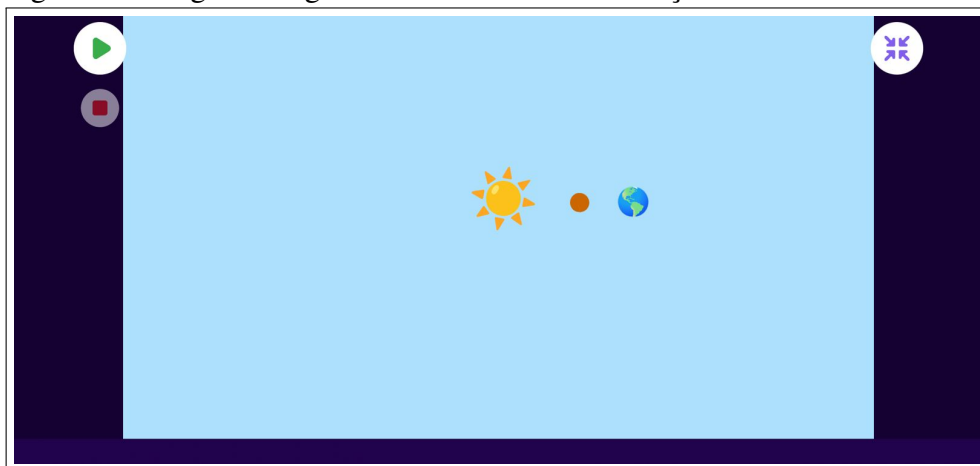
O exercício da Figura 14 consiste na programação do movimento de uma bola, que muda de direção e ângulo quando toca em uma das bordas do palco. Já o exercício mostrado na Figura 15a trata de um programa para separar a dezena e a unidade de um número aleatório. Por fim, o exercício apresentado na Figura 15b, trata de um lápis que percorre a figura geométrica da lateral que, quando tocada, vai ao centro do palco, e é percorrida pelo lápis e depois retorna ao seu lugar.

Figura 16 – Primeira sugestão de atividade de avaliação



Fonte: Elaborado pelo autor.

Figura 17 – Segunda sugestão de atividade de avaliação



Fonte: Elaborado pelo autor.

Os exercícios foram feitos pelos alunos com a ajuda do tutor, que transmitiu a tela do seu dispositivo e guiou os alunos na resolução das atividades. A primeira atividade foi realizada por completo. Já as segunda e terceira atividade foram realizadas de forma parcial. Da segunda atividade, foi realizada a parte de dezena, enquanto do terceiro exercício, foram realizadas as figuras do triângulo e do quadrado.

Ao fim da aula, foi repassada para os alunos a atividade para a aula seguinte, que serviria como avaliação dos conhecimentos adquiridos no curso. Foram propostas a continuação das atividades realizadas de forma parcial durante a aula e duas atividades lúdicas, mostradas na Figura 16, que consiste em um jogo com um lagarto que sumia e aparecia, e o usuário pontuava quando o tocava, e na Figura 17, que consistia em uma representação de planetas orbitando o sol.

#### 4.5.2.4 Aula 04 - dia 18 de novembro de 2025

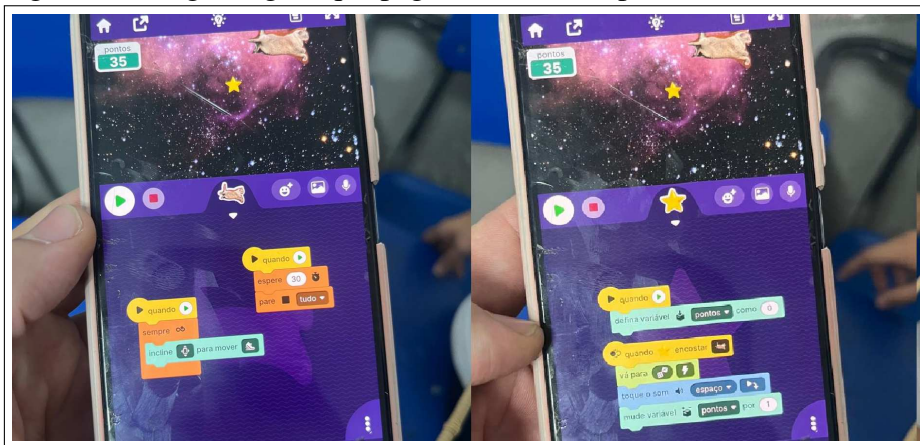
Como última aula, foi realizado o encerramento do curso com a observação dos programas realizados pelos alunos, o registro das atividades apresentadas e a anotação das habilidades demonstradas, conforme mostrado na Seção 4.3.4. Alguns registros de projetos dos alunos são apresentados na Seção 4.6.

No fim da aula, uma vez verificadas as atividades dos alunos, foram entregues os questionários de pós-curso, que foram preenchidos pelos alunos e recolhidos para registro na pesquisa.

### 4.6 Exemplos de atividades dos alunos

As Figuras 18 a 22, mostram alguns registros das atividades realizadas pelos alunos.

Figura 18 – Jogo de gato que pega estrelas feito por um aluno

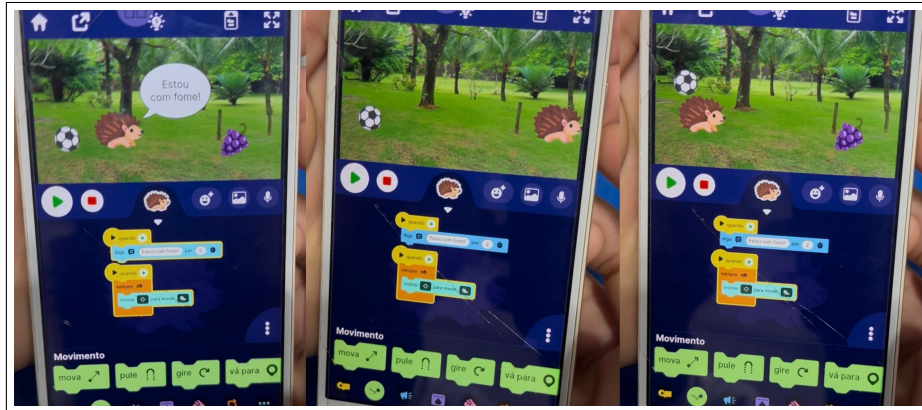


Fonte: Elaborado pelo autor.

O jogo apresentado na Figura 18 que consiste em uma atividade com movimentação por inclinação do celular, ao mudar a inclinação do celular o gato se movimenta pelo palco e, ao tocar na estrela, a variável de pontos é incrementada.

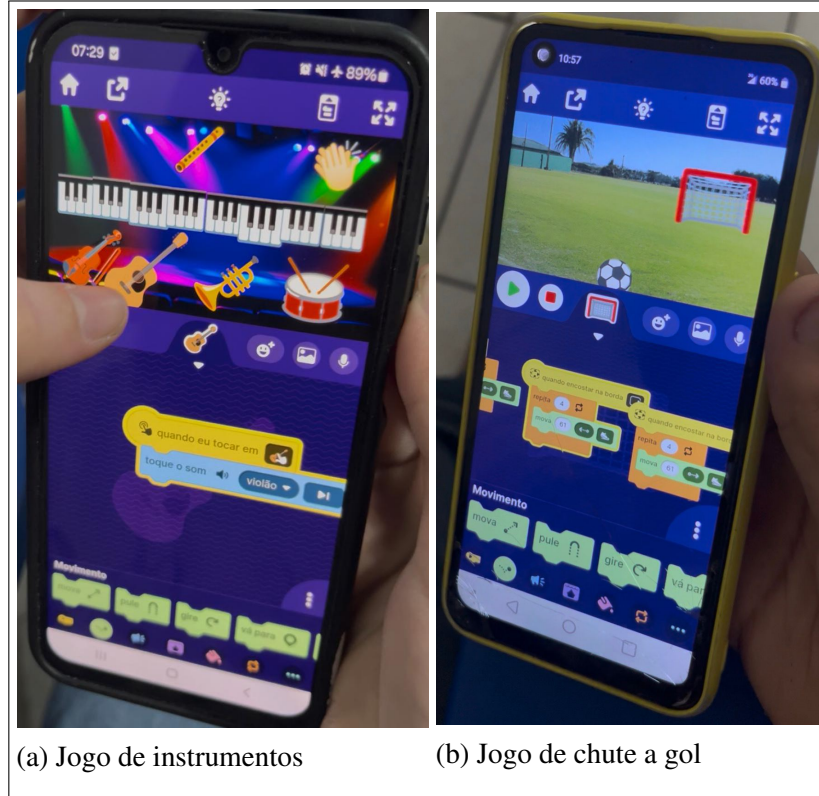
A Figura 19 apresenta um jogo de movimentação por inclinação do celular em que o ator, o porco-espinho, se movimenta na vertical conforme o celular é inclinado para os lados. Quando o ator toca um dos outros atores no palco, isto é, a uva e a bola, o porco-espinho e estes atores realizam uma ação.

Figura 19 – Programa de movimentação de um porco-espinho feito por um aluno



Fonte: Elaborado pelo autor.

Figura 20 – Jogos de instrumentos e chute a gol feitos por alunos



(a) Jogo de instrumentos

(b) Jogo de chute a gol

Fonte: Elaborado pelo autor.

A atividade apresentada na Figura 20a consiste em atores instrumentos e palmas que tocam sons associados a eles quando tocados pelo usuário. Já a Figura 21b apresenta uma atividade de chute a gol na qual o ator bola se movimenta na horizontal, em direção ao gol, e o ator trave se movimenta na vertical de um lado ao outro do palco.

A Figura 21a apresenta um jogo em que o ator panda se movimenta verticalmente conforme o celular é inclinado e deve pegar os atores alimentos, que se movimentam na horizontal, ficando abaixo deles, e evitar os atores objetos, que se movimentam da mesma forma.

Figura 21 – Jogo de pegar comidas com panda e movimentação por cenários feito por um aluno



Fonte: Elaborado pelo autor.

O programa apresentado na Figura 21b mostra um ator que se movimenta para a direita conforme o ator seta, no canto inferior direito, é tocado e interage com outros atores no palco, quando toca estes atores. Ao chegar ao fim do palco no lado direito, o cenário é alterado, reforçando e impressão de movimento.

Figura 22 – Jogo de saltar sobre obstáculos feito por um aluno



Fonte: Elaborado pelo autor.

O projeto apresentado na Figura 22 mostra um jogo onde o ator dinossauro salta sobre os obstáculos de cacto, incrementando a variável pontos e encerrando ao encostar em algum dos cactos.

#### **4.7 Considerações finais**

O conjunto de escolhas metodológicas, especialmente a respeito da investigação do uso do OctoStudio como ferramenta para o ensino de Pensamento Computacional no Ensino Fundamental e a opção por uma abordagem qualitativa, aliada à aplicação direta de atividades com os alunos, permitiu compreender não apenas os resultados objetivos, mas também as percepções subjetivas dos envolvidos.

A seleção dos instrumentos, como os questionários e avaliação das produções dos estudantes, garantiu a coleta de dados ricos e contextualizados, capazes de revelar informações importantes do processo de aprendizagem. Além disso, a descrição dos procedimentos adotados assegura a possibilidade de replicação do estudo por outros pesquisadores interessados na temática.

Portanto, a metodologia delineada apresenta coerência interna, alinhamento com as diretrizes teóricas e capacidade de oferecer subsídios concretos para a análise crítica do uso do OctoStudio no contexto escolar. Essa estrutura fornece bases sólidas para a interpretação dos resultados e para a formulação de recomendações futuras.

Nesta seção, foram apresentados os materiais e os métodos utilizados na realização da pesquisa. Todas as aulas se deram de maneira presencial, ministradas pelo tutor e autor desta pesquisa e acompanhadas pela professora da turma. A turma apresentou interesse pelo aplicativo e pelas atividades realizadas, demonstrando o potencial da ferramenta para incentivar o engajamento dos alunos no aprendizado de Pensamento Computacional e no reforço dos conteúdos da disciplina de matemática através de exercícios lúdicos e criativos. Detalhes sobre os resultados obtidos com as avaliações, são mostrados na próxima seção.

## 5 RESULTADOS

### 5.1 Introdução

Esta seção apresenta os resultados obtidos a partir dos dados coletados nos questionários mencionados na Seção 4.4 e da avaliação mencionada na Seção 4.3.4. Este trabalho teve como objetivo principal avaliar o uso do aplicativo OctoStudio como ferramenta para o ensino de Pensamento Computacional para uma turma do sexto ano de uma escola de Ensino Fundamental, da cidade de Limoeiro do Norte, no interior do Ceará.

As próximas subseções descrevem os dados coletados na pesquisa e identificam a relação entre os resultados obtidos com respeito à avaliação do curso, ao perfil dos alunos e ao desempenho apresentado. Foram consideradas as respostas de 24 alunos para a análise, porque apenas 24, dos 28 alunos da turma, responderam ambos os formulários.

Estas análises contribuem para a resposta da questão de pesquisa quanto ao valor do OctoStudio como ferramenta de ensino de Pensamento Computacional no contexto do ensino fundamental.

### 5.2 Questionário inicial

Foi aplicado um questionário com 20 perguntas no primeiro dia do experimento, com o objetivo de coletar informações dos alunos, suas experiências anteriores com programação, opiniões sobre programação e computação e condições de acesso a tecnologia digital e internet.

O questionário foi composto por três perguntas sociodemográficas, questionando o nome completo, a idade e o gênero do aluno, e nove perguntas de sim ou não, que são:

- a) “Você sabe o que é criar um programa de computador?”;
- b) “Você sabe fazer um programa de computador?”;
- c) “A Computação é útil no dia a dia.”;
- d) “Quero aprender computação na escola.”;
- e) “Você tem acesso à internet em casa?”;
- f) “Você tem smartphone em casa?”;
- g) “Você usa computador em casa?”;
- h) “Você já teve aulas para aprender a fazer programas de computador?”;
- i) “Você já pensou em trabalhar com computador?”.

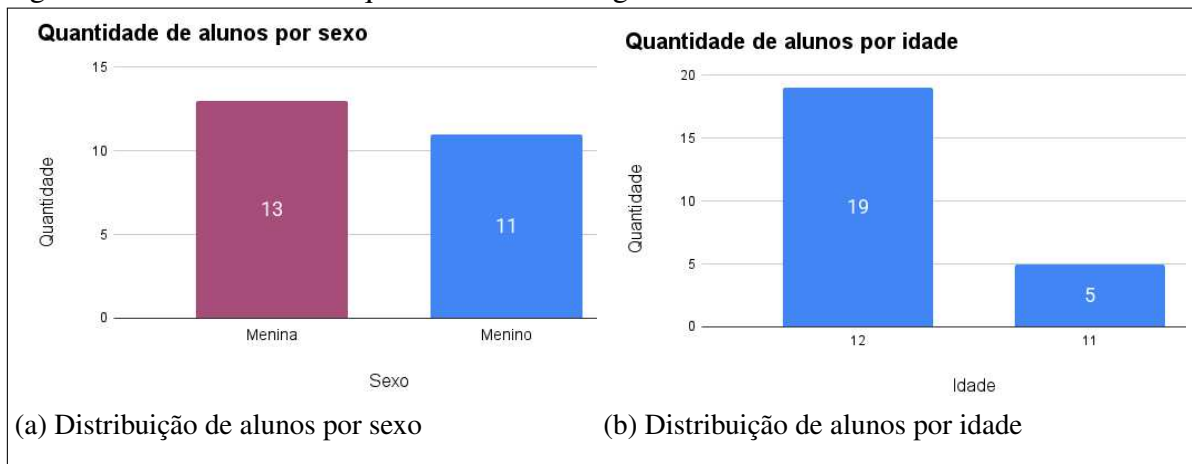
Também foram feitas as seguintes perguntas:

- a) duas perguntas de classificação;
  - “Fazer um programa de computador é:”, uma sobre a dificuldade de programar e outra sobre o grau de diversão.
- b) três perguntas fechadas sobre a quantidade de acessos semanais a internet;
- c) três perguntas sobre as atividades realizadas pelos alunos na internet.

### 5.2.1 Questões sociodemográficas

Dentre as 24 respostas dos alunos consideradas, 11 foram do sexo masculino e 13 do sexo feminino (Figura 23a), com a maioria tendo idade de 12 anos (Figura 23b).

Figura 23 – Resultados das questões sociodemográficas



Fonte: Elaborado pelo autor.

### 5.2.2 Opiniões e experiências anteriores

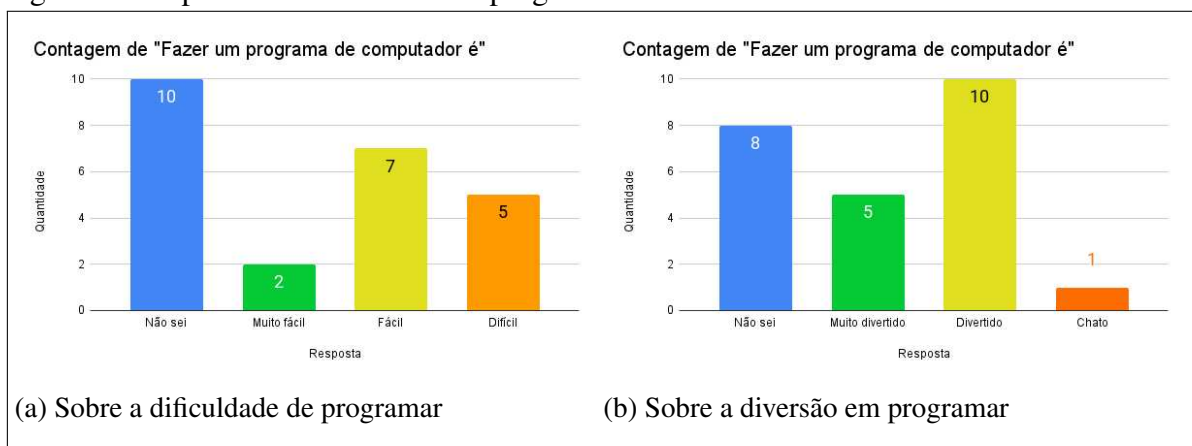
Quanto às perguntas sobre a opinião dos alunos e seus conhecimentos anteriores a respeito de programação e computação. Para a pergunta "Você sabe o que é criar um programa de computador?", 13 alunos responderam que não, enquanto 11 responderam que sim, como mostra a Figura 24.

Figura 24 – Alunos que sabem o que é criar um programa de computador



Fonte: Elaborado pelo autor.

Figura 25 – Opinião dos alunos sobre programar



Fonte: Elaborado pelo autor.

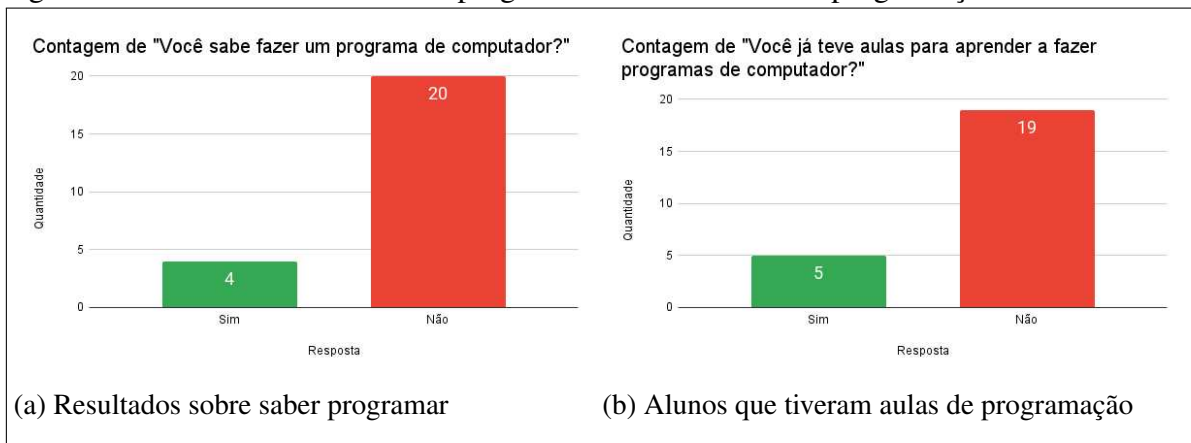
Os dados coletados e apresentados nas Figuras 24 e 25 revelam um cenário de desconhecimento inicial mitigado por uma percepção positiva quanto à natureza da atividade de programar. Conforme observado na Figura 25, a maioria dos respondentes declarou não saber o que consiste a criação de um programa de computador, o que estabelece uma linha de base de um público majoritariamente leigo no assunto.

Entretanto, ao analisar as expectativas e opiniões desses alunos, nota-se uma inclinação favorável à disciplina. No que tange à dificuldade (Figura 25a), embora o maior grupo ainda se declare indeciso (“Não sei”, com 10 menções), entre aqueles que emitiram opinião, predomina a visão de que programar é “Fácil” (sete alunos), em detrimento de uma percepção de alta complexidade (“Muito difícil”, com apenas duas menções).

O dado mais expressivo, contudo, reside na dimensão do engajamento (Figura 25b). A atividade é amplamente associada a sentimentos positivos: 15 alunos classificaram a programação entre “Divertido” ou “Muito divertido”, enquanto apenas um aluno a rotulou como "Chato". Conclui-se, portanto, que apesar da falta de domínio técnico prévio, há uma baixa resistência psicológica e um alto potencial de aceitação para a introdução do pensamento computacional no currículo deste grupo.

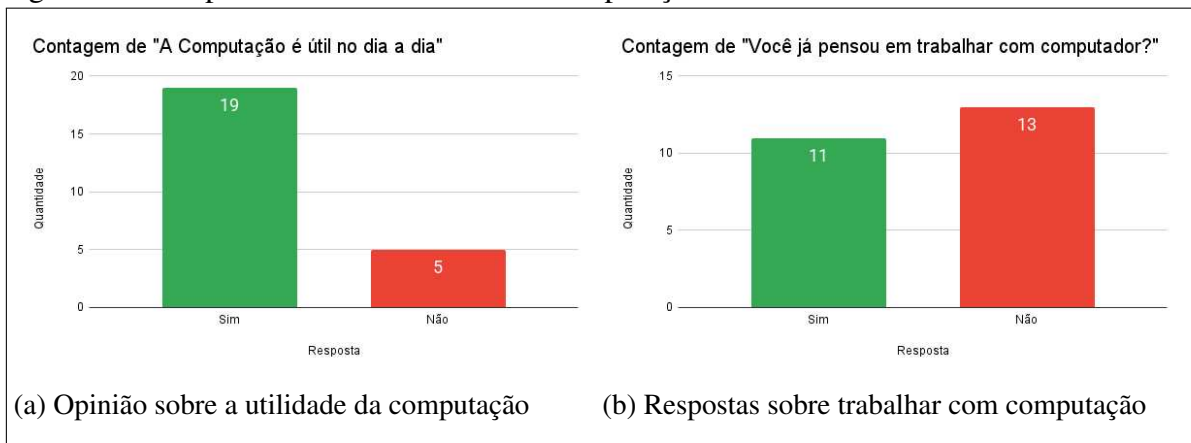
Para a pergunta “Você sabe fazer um programa de computador?”, 20 dos alunos responderam que não, enquanto quatro responderam sim (Figura 26a). Para a pergunta “Você já teve aulas para aprender a fazer programas de computador?”, cinco alunos responderam que sim, dos quais quatro era os mesmo que responderam sim para a pergunta anterior, enquanto os demais responderam não (Figura 26b).

Figura 26 – Resultados sobre saber programar e ter tido aulas de programação



Fonte: Elaborado pelo autor.

Figura 27 – Respostas sobre a utilidade da computação e seu interesse em trabalhar com ela



Fonte: Elaborado pelo autor.

Com base nos dados apresentados nas Figuras 24 a 26, os alunos apresentam um

conhecimento superficial sobre programação com 11, dos 24 alunos, tendo respondido saber o que é criar um programa de computador, ainda com divergências sobre a dificuldade e o grau de divertimento de programar. Além disso, as Figuras 26 e 27, mostram que a maioria dos alunos não sabiam programar, mas sabem sobre a utilidade da computação no dia-a-dia.

Já para a pergunta “Você já pensou em trabalhar com computador?”, 11 dos alunos responderam que sim, já haviam pensado, enquanto 13 responderam que não (Figura 27b). Por fim, para a afirmação “Quero aprender computação na escola”, todos responderam que sim, concordando com a afirmação e demonstrando o interesse dos alunos por aprender sobre programação.

Os resultados expostos na Figura 26 evidenciam uma lacuna educacional: a carência de domínio técnico (83% dos alunos não sabem programar) reflete diretamente a ausência de oferta da disciplina, visto que 79% jamais tiveram aulas sobre o tema.

Contudo, essa falta de contato não se traduz em indiferença. A Figura 27a demonstra que a utilidade da computação no cotidiano é reconhecida por quase a totalidade da amostra (19 alunos). Esse reconhecimento reflete-se no interesse profissional, onde aproximadamente 45% dos discentes já cogitaram carreiras na área tecnológica, indicando um mercado potencial que carece apenas de estímulo e instrução formal para se desenvolver

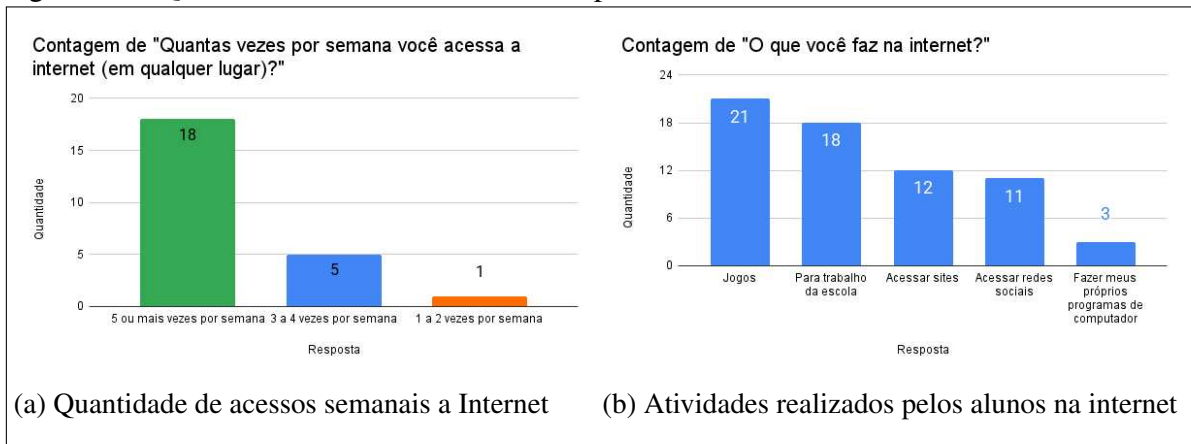
### 5.2.3 Acesso à internet e tecnologia

Para as perguntas sobre acesso à internet e tecnologia, todos os alunos declararam ter acesso a internet em casa, e para a pergunta “Quantas vezes por semana você acessa a internet (em qualquer lugar)?”, 18 alunos responderam acessar cinco ou mais vezes por semana, cinco alunos responderam acessar três a quatro vezes por semana e um alunos respondeu acessar uma a duas vezes por semana (Figura 28a).

Adicionalmente, para a pergunta de múltiplas opções “O que você faz na internet?”, dos 24 alunos, 21 responderam acessar a internet para jogar, 18 para realizar trabalhos da escola, 12 acessar sites, 11 para acessar redes sociais e três responderam acessar para fazer seus próprios programas de computador (Figura 28b), e não houveram observações na área destinada para comentários desta questão.

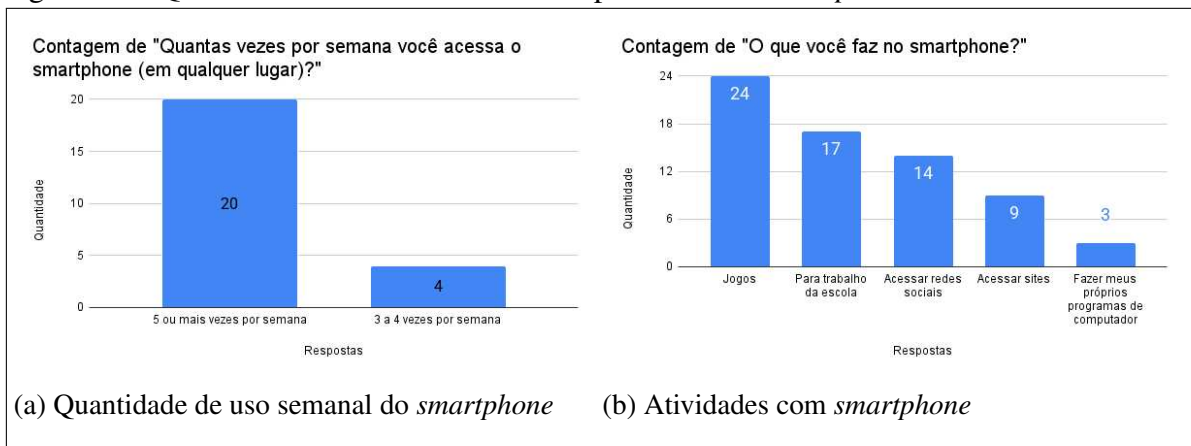
Para a pergunta “Você tem *smartphone* em casa?”, todos os alunos responderam que sim e para a pergunta “Quantas vezes por semana você acessa o *smartphone* (em qualquer lugar)?”, 20 alunos responderam acessar 5 ou mais vezes por semana e 4 alunos responderam

Figura 28 – Quantidade de acesso semanal e tipo de uso da internet



Fonte: Elaborado pelo autor.

acessar 3 a 4 vezes por semana (Figura 29a). Já para a pergunta de múltiplas escolhas "O que você faz no *smartphone*?", 24 alunos responderam usar o *smartphone* para jogos, 17 para trabalhos escolares, 14 para acessar redes sociais, nove para acessar sites e três responderam usar para fazer seus próprios programas (Figura 29b).

Figura 29 – Quantidade de acesso semanal e tipo de uso do *smartphone*

Fonte: Elaborado pelo autor.

Ao analisar os hábitos digitais e a formação técnica dos alunos, observa-se um perfil de "nativo digital consumidor". Embora a grande maioria acesse a internet e smartphones cinco ou mais vezes por semana, esse uso é predominantemente voltado ao entretenimento, como jogos e redes sociais.

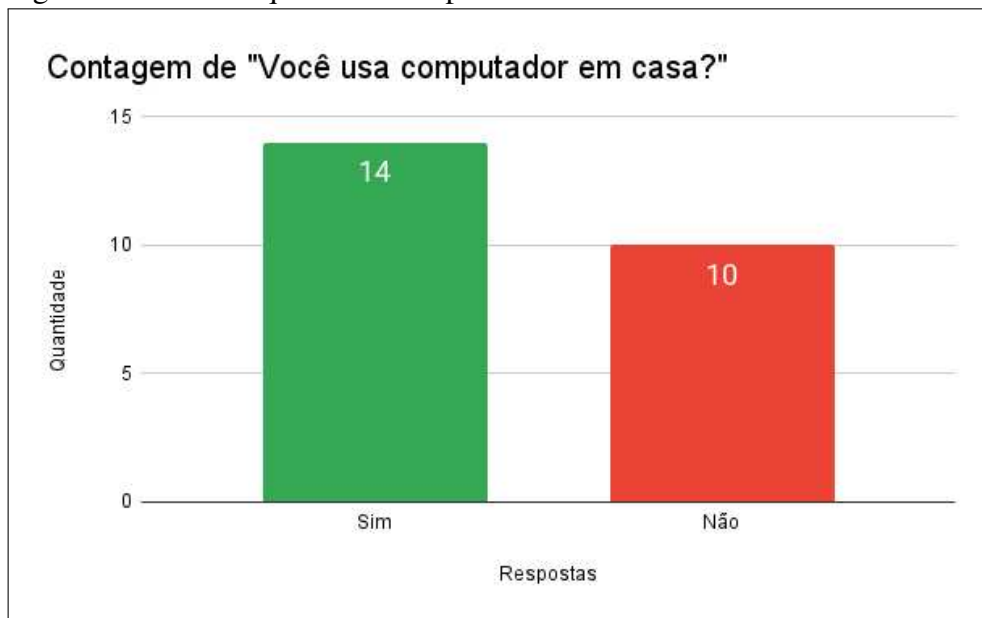
Esse comportamento reflete a carência de educação tecnológica formal evidenciada na Figura 26, onde a ausência de aulas de programação (19 alunos) justifica a incapacidade técnica declarada pela quase totalidade da amostra. Paradoxalmente, existe um terreno fértil para o ensino: os alunos reconhecem a utilidade da computação (19 menções) e quase metade

demonstra interesse em seguir carreira na área, indicando que a transição de usuários passivos para criadores ativos depende primordialmente da oferta de instrução técnica

Sobre o uso de computador, 14 alunos responderam usar computador em casa e 10 responderam não usar (Figura 30). Deste, os mesmo 10 responderam nunca usar o computador, um respondeu usar 5 ou mais vezes por semana, dois responderam usar o computador 3 a 4 vezes por semana e 11 responderam usar 1 a 2 vezes por semana (Figura 31a).

Sobre as atividades realizadas no computador, 13 alunos responderam usar o computador para fazer trabalhos escolares, oito alunos responderam usar para acessar sites, sete para jogar, um alunos respondeu usar para fazer seus próprios programas e 10 responderam não usar o computador (Figura 31b).

Figura 30 – Alunos que usam computador

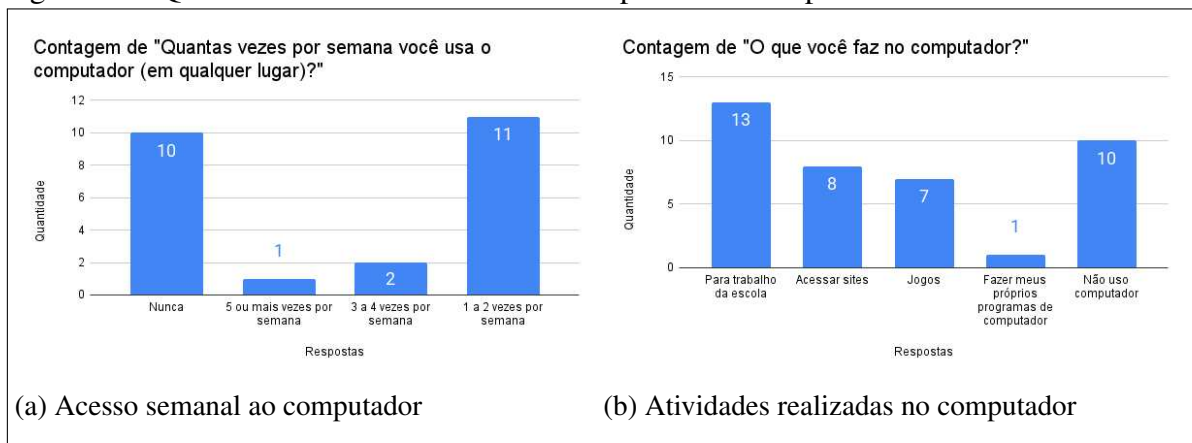


Fonte: Elaborado pelo autor.

Os resultados evidenciam um contraste significativo entre o acesso tecnológico e o letramento computacional dos estudantes. A turma apresentou uma discrepância entre o acesso a computadores e a *smartphones*, pois todos declararam ter acesso a *smartphones* mas apenas 14 declararam ter acesso a computador, e ainda com o uso bem menos frequente do computador.

Complementarmente, as atividades realizadas em cada aparelho também mostraram-se distintas em “formalidade”, com o computador sendo o mais acessado para realizar trabalhos para escola, mas o *smartphone* para jogos e entretenimento. Assim, a apresentação de um novo uso para *smartphone* mostra-se relevante.

Figura 31 – Quantidade de acessos semanais e tipo de uso computador



Fonte: Elaborado pelo autor.

### 5.3 Questionário final

Ao final do curso, foi aplicado o questionário final com 17 perguntadas adaptadas do modelo dTECT, com perguntas abertas e fechadas e espaços para comentário com o objetivo de avaliar, na percepção dos estudantes, o interesse deles no curso e sua qualidade.

O questionário foi composto pela seguintes perguntas:

- a) uma pergunta do nome completo do alunos;
- b) três perguntas abertas:
  - “O que menos gostei nas aulas foi.”;
  - “O que mais gostei nas aulas foi.”;
  - “O que você achou destas aulas?”.
- c) sete perguntas de sim ou não:
  - “Eu consigo programar”;
  - “Mostrei o programa que fiz no smartphone para outras pessoas”;
  - “Consigo explicar para um amigo/amiga como programar”;
  - “Quero aprender mais sobre programar”;
  - “Eu gosto de programar”;
  - “A Computação é útil no dia-a-dia.”;
  - “Você já pensou em trabalhar com computação?”.
- d) seis perguntas de múltipla escolha:
  - “As aulas foram:”, para classificação da qualidade geral do curso, sua dificuldade e grau de diversão com as aulas.
- e) duas perguntas sobre programação:

– “Programação é:”, para classificar a dificuldade e grau de diversão com programação.

f) uma pergunta sobre a percepção do tempo decorrido das aulas.

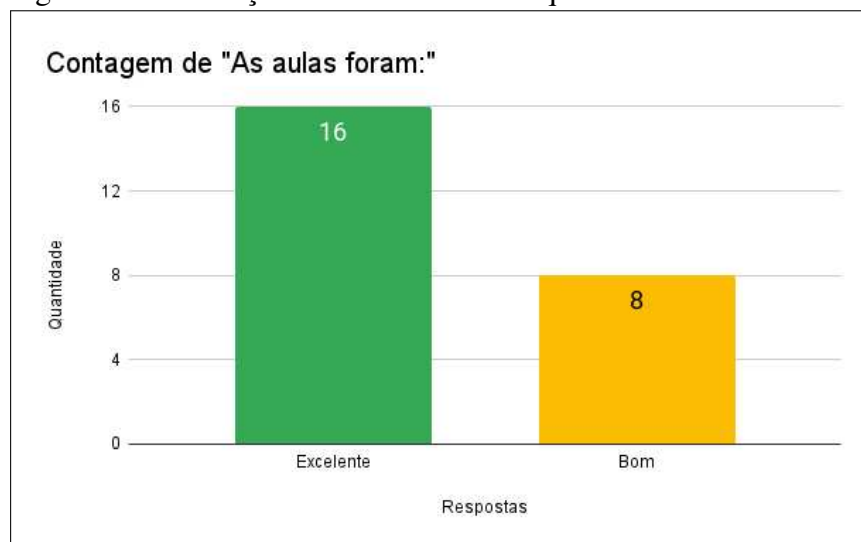
### 5.3.1 Questões sobre as aulas

Foram feitas quatro perguntas classificatórias sobre a opinião dos alunos a respeito das aulas, três consistiam em questões fechadas, com o enunciado “As aulas foram”, com opções para marcar o que eles achavam sobre a qualidade, a dificuldade e o grau de diversão das aulas, e uma com o enunciado “O tempo das aulas passou”, com opções para marcar a percepção da passagem do tempo das aulas.

Para a pergunta “As aulas foram” com foco no grau de dificuldade, que tinha as opções de resposta “Muito fáceis”, “Fáceis”, “Difíceis” e “Muito difíceis”, foram obtidas duas respostas para muito fáceis, 20 para fáceis, uma para difíceis, onde foi deixado um comentário “Mais ou menos” nesta resposta, e uma para muito difíceis (Figura 37a).

Já para a pergunta “As aulas foram” com foco na qualidade geral do curso, com as opções “Excelente”, “Bom”, “Regular” e “Ruim”, foram obtidas 16 respostas para opção excelente e oito respostas para bom (Figura 32).

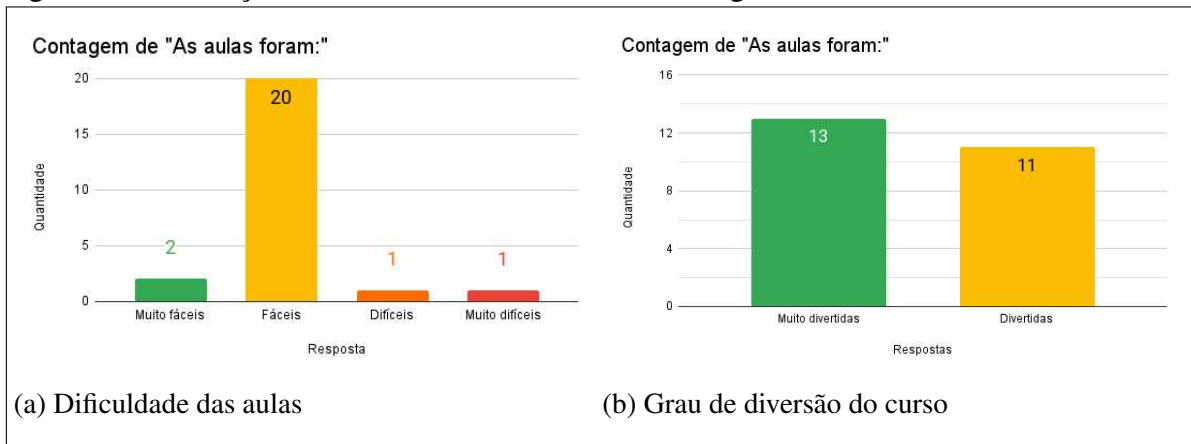
Figura 32 – Avaliação dos alunos sobre a qualidade do curso



Fonte: Elaborado pelo autor.

Para a classificação sobre a diversão das aulas com a pergunta “As aulas foram” focada neste aspecto, com as opções “Muito divertidas”, “Divertidas”, “Chatas” e “Muito chatas”, foram obtidas 13 respostas para muito divertidas e 11 respostas para divertidas (Figura 37b).

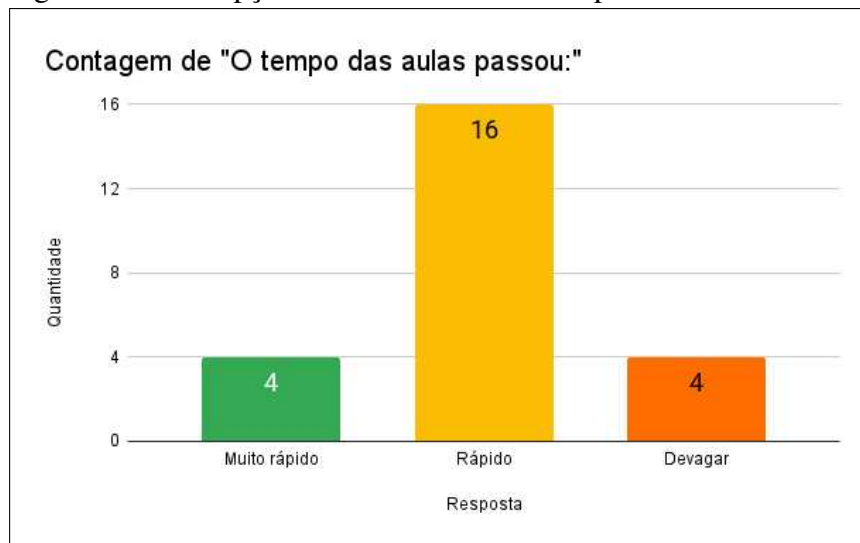
Figura 33 – Avaliação dos alunos sobre a dificuldade e o grau de diversão das aulas



Fonte: Elaborado pelo autor.

Por fim, a pergunta “O tempo das aulas passou”, classificável entre “Muito rápido”, “Rápido”, “Devagar” e “Muito devagar”, onde a rapidez de uma aula está associada a quanto ela prende a atenção e a lentidão, ao contrário, como resposta, a questão obteve 16 respostas como rápido, quatro como muito rápido e quatro como devagar (Figura 34).

Figura 34 – Percepção dos alunos sobre o tempo das aulas



Fonte: Elaborado pelo autor.

Os resultados obtidos após a conclusão do curso (Figuras 32 a 34) ratificam a eficácia da proposta pedagógica. A percepção de qualidade foi unânime, com 100% da turma avaliando as aulas entre 'Bom' e 'Excelente'. O dado mais relevante reside na quebra de paradigma quanto à complexidade do tema: enquanto o diagnóstico inicial apresentava insegurança, o resultado final mostrou que 91,6% dos discentes consideraram a aprendizagem 'Fácil' ou 'Muito fácil'.

Além disso, o engajamento emocional foi elevado, visto que a totalidade da amostra

classificou a experiência como divertida. Essa satisfação é corroborada pela percepção temporal, onde 83,3% dos alunos relataram que as aulas transcorreram de forma rápida ou muito rápida, evidenciando um ambiente de aprendizagem dinâmico e motivador.

Ainda sobre as aulas, na pergunta aberta sobre o que mais gostaram das aulas, os alunos escreveram:

- a) “de tudo”;
- b) “eu achei muito criativo”;
- c) “Podendo fazer minhas criações”;
- d) “tudo”;
- e) “aprender sobre o aplicativo”;
- f) “Pode fazer sua história do seu jeitinho”;
- g) “A forma de mexer e interagir com o personagem”;
- h) “O uso do aplicativo”;
- i) “aprender a como usar o aplicativo”;
- j) “A forma que foi explicado”;
- k) “programar”;
- l) “O que mais gostei foi aprender a mexer no aplicativo pois é muito legal”;
- m) “tudo”;
- n) “de fazer jogos”;
- o) “De fazer jogos divertidos e criativos ”;
- p) “a programação de como fazer jogos”;
- q) “dos jogos”;
- r) “Fazer o programa com meus amigos ”;
- s) “Gostei de trazer o celular e as aulas ”;
- t) “foram as partes que estávamos aprendendo todos juntos”;
- u) “O desenvolvimento”;
- v) “Pode aprender sobre tecnologia”.

Os comentários podem ser classificados em algumas categorias como: “tudo”, com três comentários elogiando o curso de forma geral; “criatividade”, com quatro comentários elogiando as formas de criar histórias e interagir com os personagens; “aplicativo”, com cinco elogios à tecnologia e o aprendizado do aplicativo; “desenvolvimento de jogos”, com sete comentários sobre programação dos jogos; e “aspectos diversos das aulas” com três comentários

sobre a explicação das aulas, o uso dos celulares e o aprendizado junto com os colegas.

Com base nos comentários, com a maioria deles sobre o desenvolvimento de jogos, o aplicativo e as possibilidades criativas permitidas por ele, os alunos demonstraram seu interesse pelo método utilizado nas aulas com o uso do aplicativo e com as atividades lúdicas. O engajamento dos alunos é corroborado pelos resultados demonstrados pelos gráficos da Figura 37a, que destaca as aulas como fáceis, Figura 32, que classifica as aulas como excelentes e boas e Figura 37b, que as descreve como muito divertidas e divertidas.

Já para a pergunta sobre o que menos gostaram nas aulas, os alunos responderam:

- a) “não tem”;
- b) “nada”;
- c) “quando acaba a aula”;
- d) “nada”;
- e) “amei tudo”;
- f) “por que passou muito rápido”;
- g) “Porque passou muito rápido”;
- h) “nada”;
- i) “muita coisa para aprender”;
- j) “o barulho [dos colegas]”;
- k) “nada”;
- l) “nada”;
- m) “que acabou”;
- n) “quando acabou”;
- o) “quando acabava e a gente ia embora para casa”;
- p) “gostei de todas”;
- q) “Não gostei quando acabou ”;
- r) “O barulho dos meus colegas ”.

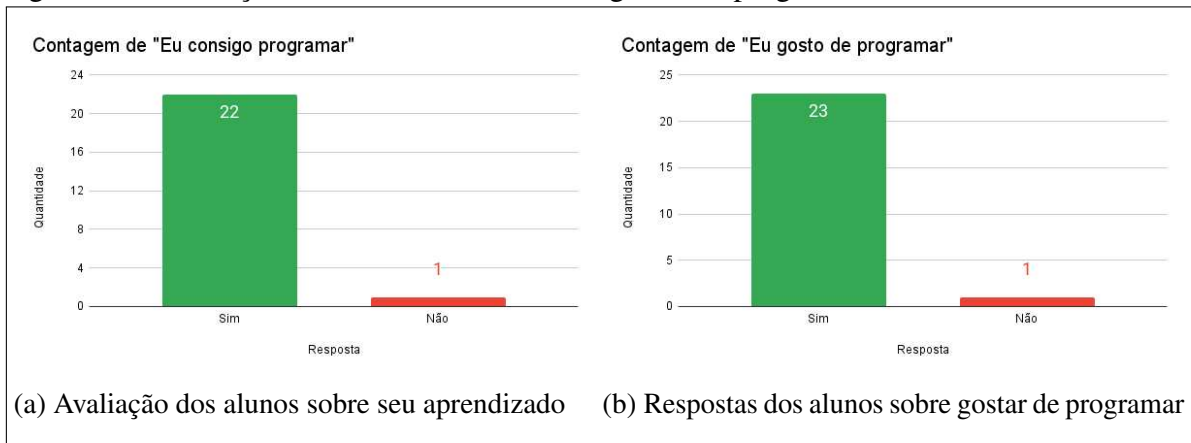
As respostas sobre o que menos gostaram nas aulas podem ser organizadas em algumas categorias, como: “gostei de tudo”, com oito respostas como “gostei de tudo” ou “nada” no sentido de que não houve nada que menos gostaram; “quando acabou a aula”, com cinco comentário indicando que os alunos gostariam de ter mais aulas sobre a ferramenta; “duração do curso”, com dois comentários sobre o curso ter passado muito rápido; e “diversos” três comentários sobre o comportamento da turma e sobre a quantidade de conteúdo a ser aprendido.

Com base nos comentários sobre o que menos gostaram, pode-se notar uma aprovação do curso por parte dos alunos, tendo a maioria dos comentários declarado que não havia nada que não gostaram e não terem gostado quando as aulas acabaram, o que mostrou o interesse deles em ter mais aulas de programação.

### 5.3.2 Questões sobre aprendizado e envolvimento

Para a pergunta de autoavaliação “Eu consigo programar”, foram obtidas 23 respostas porque um dos alunos deixou esta questão em branco. Dentre as respostas, 22 foram marcadas como “Sim”, com uma delas acompanhada de uma comentário “mais ou menos”, e uma resposta “Não”, como mostra a Figura 35a.

Figura 35 – Avaliação dos alunos sobre saber e gostar de programar



Fonte: Elaborado pelo autor.

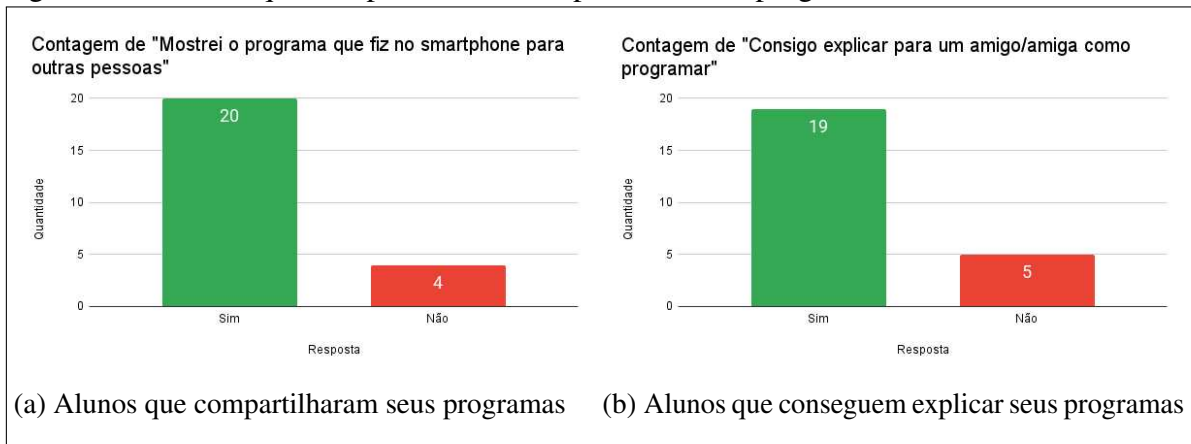
A questão “Mostrei o programa que fiz no smartphone para outras pessoas” recebeu 20 respostas “Sim” e quatro respostas “Não” (Figura 36a). A questão “Consigo explicar para um amigo/amiga como programar” recebeu 19 respostas “Sim” e cinco respostas “Não” (Figura 36b).

A intervenção pedagógica atingiu objetivos que transcendem a mera alfabetização técnica. Enquanto os dados iniciais apontavam uma lacuna severa de instrução (79

A eficácia do método reflete-se na segurança cognitiva dos estudantes: 19 deles declararam-se aptos a atuar como multiplicadores do conhecimento, explicando a lógica de programação para seus pares. Aliado ao alto índice de satisfação e diversão relatado, conclui-se que o curso não apenas ensinou uma nova habilidade, mas fomentou o protagonismo digital, convertendo o interesse latente em competência real e prazerosa.

A questão “Eu gosto de programar” recebeu 23 respostas “Sim” e uma resposta

Figura 36 – Alunos que compartilharam e explicaram seus programas



Fonte: Elaborado pelo autor.

“Não” (Figura 35b). Algo interessante foi que o único aluno que respondeu não conseguir programar, respondeu “Sim” a esta pergunta, enquanto o único “Não” desta pergunta veio de um aluno que respondeu que conseguia programar. Por fim, a questão “Quero aprender mais sobre programação” foi marcada como “Sim” por todos os alunos.

A análise evolutiva dos dados demonstra um impacto transformador da intervenção de ensino de programação. Inicialmente, o cenário era de amplo desconhecimento técnico (83% de leigos) e um uso de tecnologia estritamente passivo e voltado ao entretenimento. Contudo, após o curso, os índices de autoeficácia sofreram uma mudança radical: a grande maioria dos discentes (22 alunos) passou a declarar-se capaz de programar.

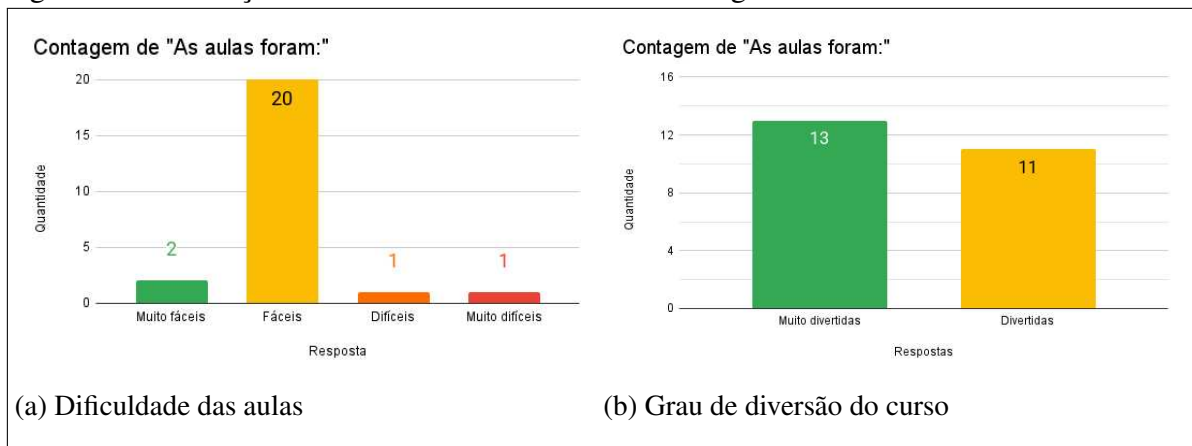
A intervenção pedagógica atingiu objetivos que transcendem a mera alfabetização técnica. Enquanto os dados iniciais apontavam uma lacuna severa de instrução (79% nunca tiveram aulas), os resultados finais demonstram uma transformação profunda na relação do aluno com a tecnologia. A transição do uso passivo do smartphone (anteriormente focado em jogos) para um uso produtivo e social é evidenciada pelo fato de 20 alunos terem compartilhado suas criações autorais com terceiros.

A eficácia do método reflete-se na segurança cognitiva dos estudantes: 19 deles declararam-se aptos a atuar como multiplicadores do conhecimento, explicando a lógica de programação para seus pares. Aliado ao alto índice de satisfação e diversão relatado, conclui-se que o curso não apenas ensinou uma nova habilidade, mas fomentou o protagonismo digital, convertendo o interesse latente em competência real e prazerosa.

### 5.3.3 Questões sobre programação e computação

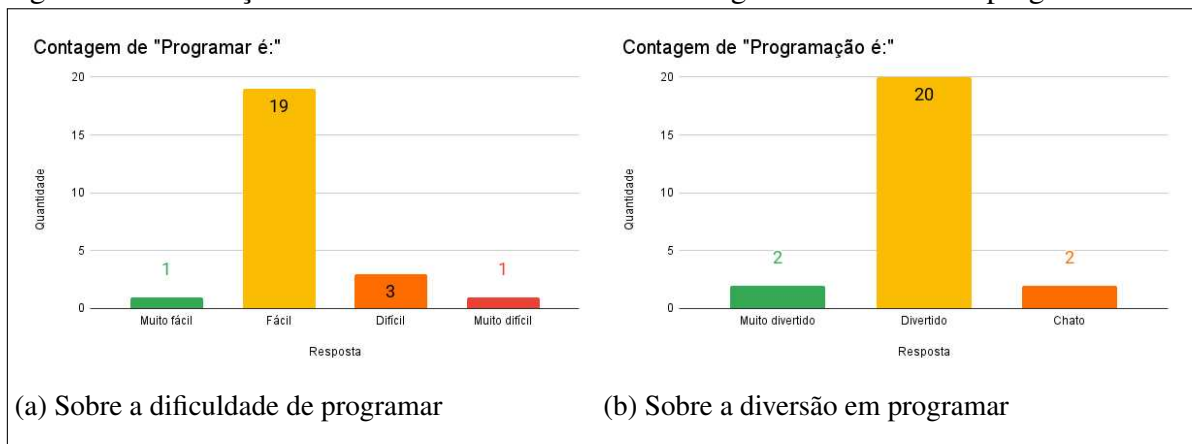
A questão “Programar é”, que identifica como os alunos classificam a dificuldade de programar, com as opções “Muito fácil”, “Fácil”, “Difícil” e “Muito difícil”, recebeu uma resposta como “Muito fácil”, 19 respostas como “Fácil”, três respostas como “Difícil” e uma resposta como “Muito difícil” (Figura 38a).

Figura 37 – Avaliação dos alunos sobre a dificuldade e o grau de diversão das aulas



Fonte: Elaborado pelo autor.

Figura 38 – Avaliação dos alunos sobre a dificuldade e o grau de diversão de programar



Fonte: Elaborado pelo autor.

Já para a questão “Programação é”, voltada para levantar o quanto os alunos consideraram divertido programar, com as opções “Muito divertido”, “Divertido”, “Chato” e “Muito chato”, obteve duas resposta como “Muito divertido”, 20 respostas como “Divertido” e duas respostas como “Chato” (Figura 38b).

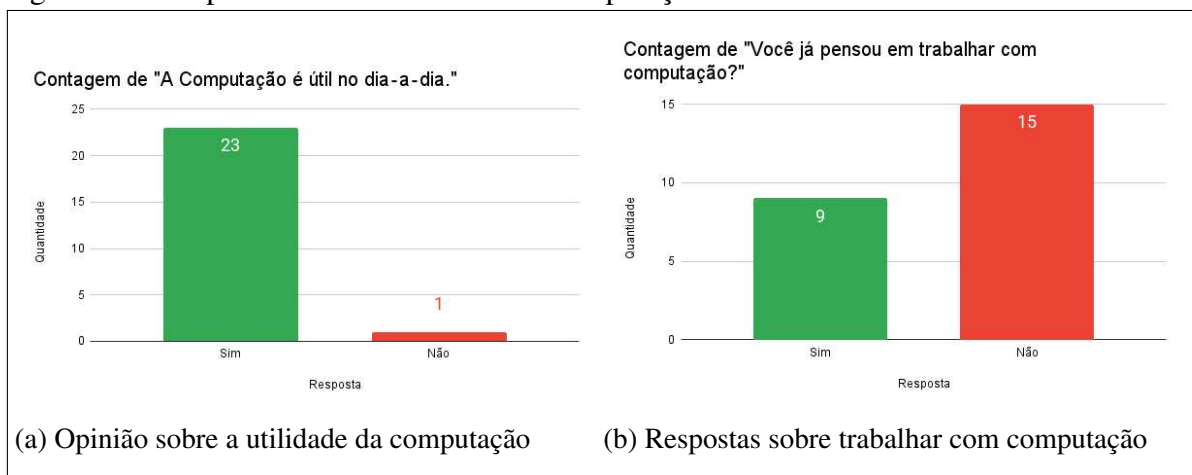
A análise dos dados permite concluir que a intervenção pedagógica não apenas capacitou os alunos tecnicamente, mas também promoveu uma ressignificação da programação.

O que antes era um conceito abstrato e potencialmente difícil tornou-se, para 83,3% da turma, uma atividade de fácil execução.

Além disso, o alto índice de ludicidade observado demonstra que a barreira emocional em relação às ciências exatas e tecnológicas foi superada. Esse entusiasmo, aliado à capacidade declarada de explicar o conteúdo a terceiros (Figura 36b), confirma que o curso cumpriu seu papel de introduzir o pensamento computacional de forma eficaz, prazerosa e socialmente compartilhável.

A questão “A Computação é útil no dia-a-dia” recebeu 23 respostas “Sim” e uma resposta “Não” (Figura 39a), o que mostra um aumento de respostas “Sim” em relação aos resultados do questionário inicial. Já a pergunta “Você já pensou em trabalhar com computação?”, mostrou uma diminuição de respostas “Sim” com relação a mesma pergunta no questionário inicial, tendo nove respostas “Sim” e 15 respostas “Não” (Figura 39b).

Figura 39 – Respostas sobre a utilidade da computação e seu interesse em trabalhar com ela



Fonte: Elaborado pelo autor.

Ao comparar-se os dados sobre o interesse em trabalhar com computação (Figura 39b) e as respostas sobre o interesse em aprender mais sobre programação, sobre a utilidade da computação, o interesse pelo curso e a satisfação em programar, a queda nas respostas sim sobre querer trabalhar com computação poder ser justificada por não ter sido trabalhado diretamente com computador de mesa durante o curso e os alunos terem associado a pergunta com ele, uma outra formulação da pergunta poderia ser sido sobre trabalha com programação em si.

A partir dos dados do questionário final, pôde-se notar uma maior consistência, em comparação com o questionário inicial, sobre o que os alunos pensam sobre a dificuldade de programar e o quão divertido eles consideraram programar. Além do aumento de quatro respostas

concordando sobre a utilidade da computação no dia-a-dia. Porém, algo surpreendente foi a diminuição na quantidade de alunos que afirmam ter interesse em trabalhar com computador.

#### 5.4 Avaliação dos conhecimentos

Foram feitas notas de observações pelo tutor para cada aluno, para avaliação das habilidades de acordo com as diretrizes do BNCC apresentadas pelos alunos ao longo do curso e na apresentação da atividade final, conforme definido na Seção 4.3.4, este são:

- a) (EF06CO02) Elaborar algoritmos que envolvam instruções sequenciais, de repetição e de seleção usando uma linguagem de programação;
- b) (EF06CO03) Descrever com precisão a solução de um problema, construindo o programa que implementa a solução descrita;
- c) (EF06CO04) Construir soluções de problemas usando a técnica de decomposição e automatizar tais soluções usando uma linguagem de programação.

Assim, foi feita uma avaliação subjetiva com base nas participações dos alunos nas aulas e na apresentação das atividades propostas. Foi atribuído a cada um as qualificações “Bom”, “Razoável” e “Insuficiente”.

Um exemplo de código “Bom” é mostrado na Figura 40, com comandos organizados por personagens e eventos, como os comandos do personagem da lua, do lado direito da figura, com blocos de código organizados por eventos, laço de repetição e evento para encerrar o programa com requisitos de gatilho.

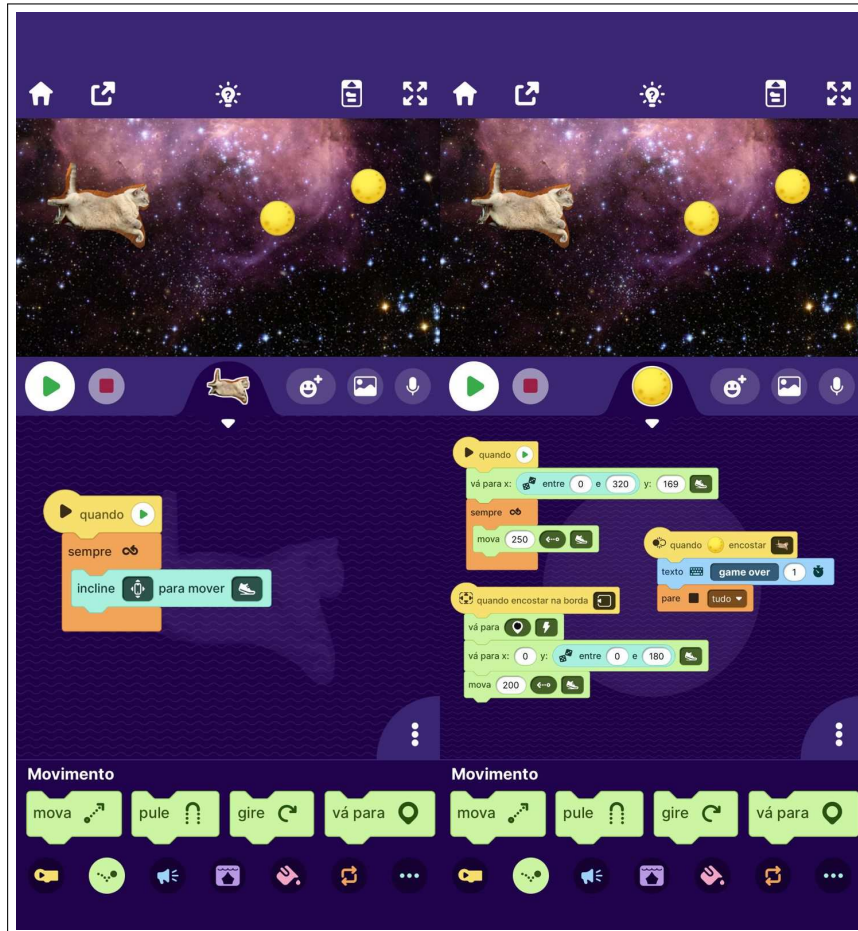
A Figura 41 apresenta um exemplo de programa “Razoável”, com comandos sucintos, eventos de gatilho limitados ou trocados, como no caso do personagem da bola, na imagem esquerda, onde seria melhor o evento de gatilho de toque no personagem, ao invés do evento de toque no botão iniciar.

Já os códigos classificáveis como “Insuficiente” limitavam-se a, no máximo, apenas comandos avulsos espalhados que, apesar de realizarem ações, não seguem uma sequência de comandos encadeados ou são ativados por eventos, isto é, não implementam um algoritmo.

Para o critério EF06CO02 (Figura 42), 21 alunos apresentaram desempenho como “Bom”, conseguido acompanhar as atividades em sala e implementar soluções, e três alunos apresentaram desempenho “Razoável”, tendo dificuldade em acompanhar as atividades e necessitando de ajuda em quase todo o processo de implementação.

Para o critério EF06CO03 (Figura 43), três alunos apresentaram desempenho “Bom”,

Figura 40 – Exemplo de código “Bom”

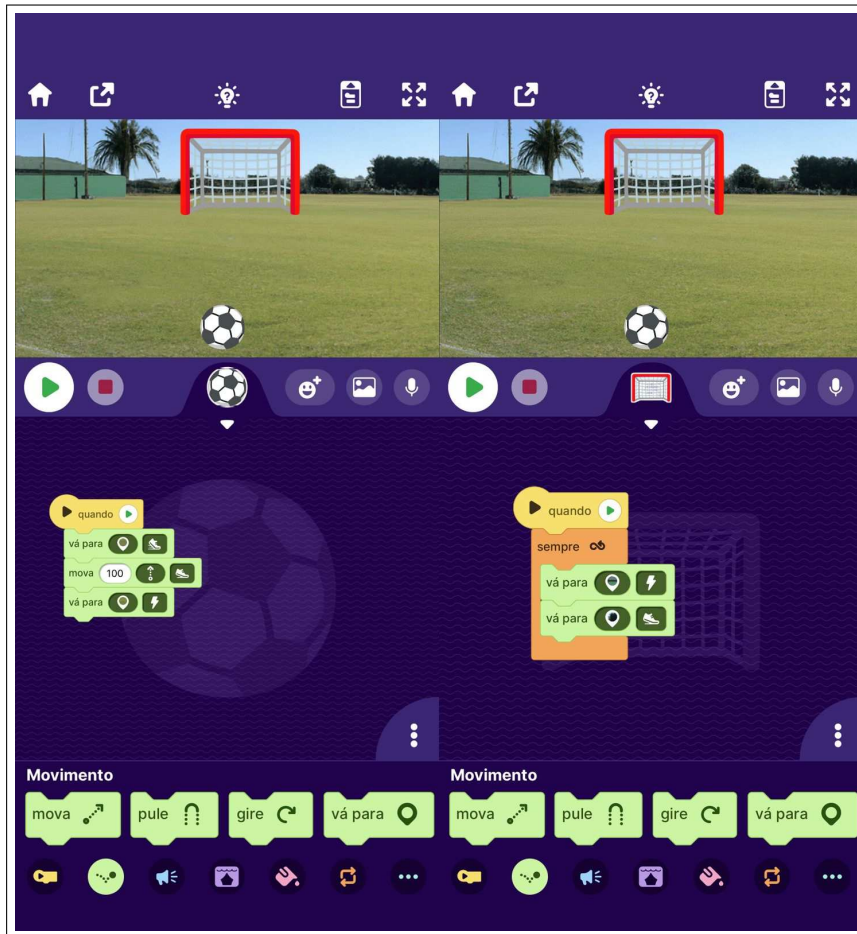


Fonte: Elaborado pelo autor.

tendo conseguido explicar sobre o que se tratavam seus programas e o que os personagens faziam com base nos comandos atribuídos para eles, e 21 apresentaram desempenho “Razoável”, por conseguirem explicar de maneira simplificada o que seus programas faziam e como seus personagens realizavam ações.

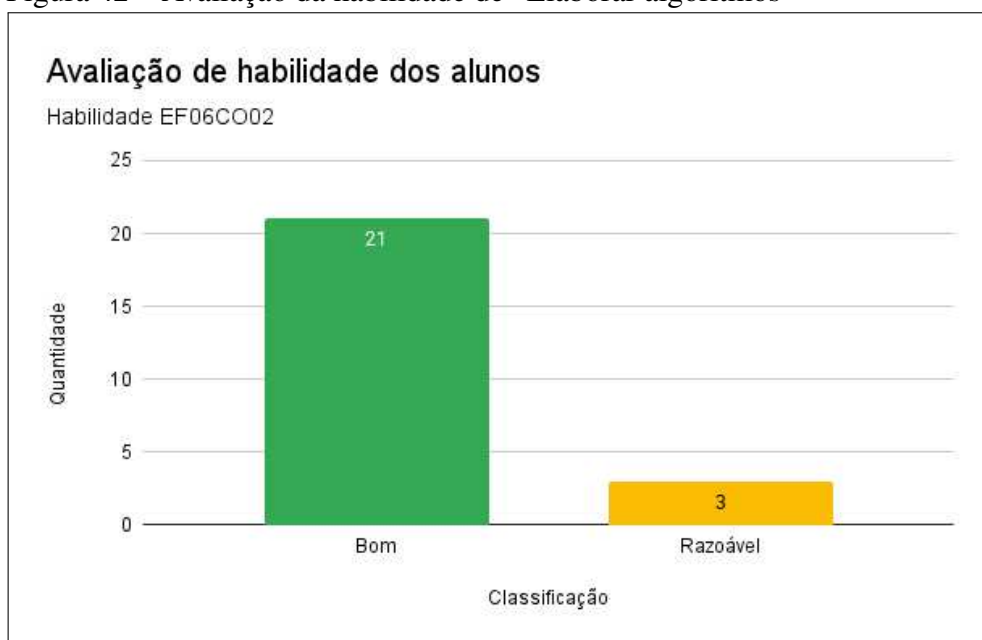
Para o critério EF06CO04 (Figura 44), com base na construção dos comandos para os personagens, seis alunos apresentaram desempenho como “Bom”, tendo implementado diversos blocos com comandos encadeados realizando ações diferentes, ao invés de deixarem os comandos em sequência, para personagens distintos nas suas soluções, e 16 alunos apresentaram desempenho “Razoável”, tendo demonstrado dificuldades em atribuir comandos para realizar as atividades propostas ou suas próprias ideias.

Figura 41 – Exemplo de código “Razoável”



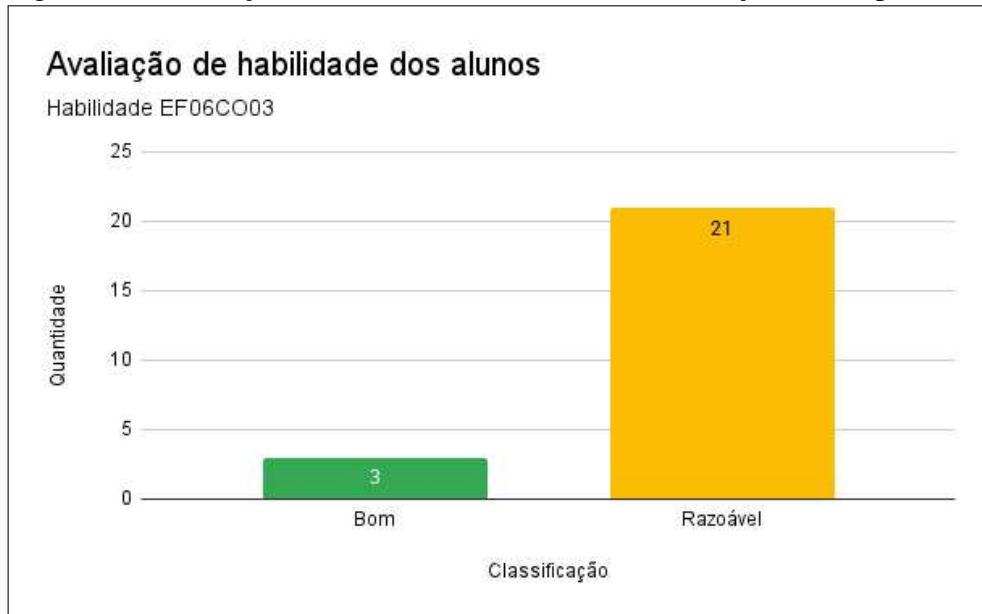
Fonte: Elaborado pelo autor.

Figura 42 – Avaliação da habilidade de “Elaborar algoritmos”



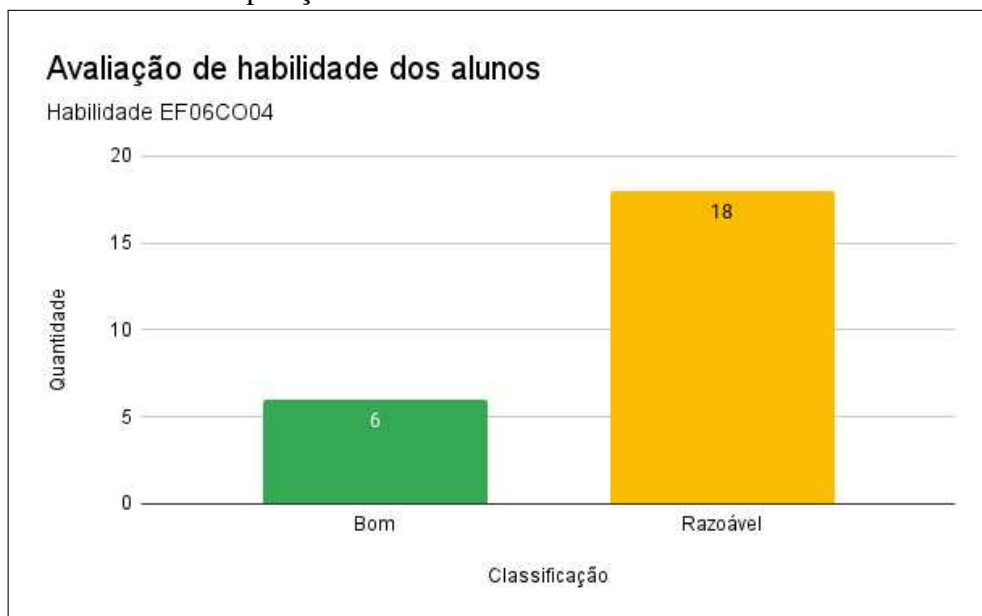
Fonte: Elaborado pelo autor.

Figura 43 – Avaliação da habilidade de “Descrever a solução de um problema”



Fonte: Elaborado pelo autor.

Figura 44 – Avaliação da habilidade de “Construir soluções automatizadas com decomposição”



Fonte: Elaborado pelo autor

## 5.5 Considerações

Com base nas respostas fornecidas pelos alunos, o aplicativo OctoStudio e este curso de aplicação foram bem aceitos pela turma. O curso foi avaliado como “excelente” e “bom” (Figura 32), a maioria dos alunos classificou as aulas como “fáceis” (Figura 37a), como “muito divertidas” e “divertidas” (Figura 37b) e que o tempo da aula foi “muito rápido” ou “rápido” (Figura 34), mostrando o engajamento por parte dos alunos às aulas, sendo corroborado pelo interesse da turma completa em aprender mais sobre programação.

Os alunos demonstraram confiança em relação a sua aprendizagem, com a maioria afirmando conseguir programar e gostar de programação, o que pôde ser visto durante as prática em sala de aula e no trabalho final. Assim, a partir dos dados coletados, a pesquisa mostrou alcançar os objetivos propostos ao captar o interesse por parte dos alunos e o aprendizado de pensamento computacional, com atividades lúdicas de programação e de matemática.

A análise integrada dos dados revela que a intervenção pedagógica foi o divisor de águas entre o uso meramente recreativo da tecnologia e o domínio técnico consciente. Inicialmente, o cenário era de total desconhecimento, onde o smartphone servia quase exclusivamente para jogos e redes sociais.

Após o curso, houve uma reversão notável: a percepção de que programar é “difícil” foi substituída por uma visão de acessibilidade e diversão, resultando em 22 alunos que agora afirmam ter competência para criar seus próprios programas.

Com base nas respostas finais sobre o interesse em trabalhar com computação, que apresentou uma queda com relação aos resultados coletados no questionário inicial, uma melhoria para a proposta pedagógica seria a inclusão de atividades com computador de mesa, quando a estrutura escolar permitir. Assim, outra ferramenta e perspectiva de trabalho poderia ser trabalhada com os alunos.

Mais do que o aprendizado técnico, a pesquisa demonstra um ganho na autoeficácia e na valorização da computação como ferramenta útil para a vida, consolidando a ideia de que o ensino de programação é um caminho eficaz para transformar jovens consumidores em criadores digitais confiantes e aptos a compartilhar conhecimento.

## 6 CONCLUSÃO

Esta seção apresenta uma síntese dos principais achados da pesquisa sobre o uso do OctoStudio como ferramenta de apoio ao desenvolvimento do pensamento computacional no Ensino Fundamental. Ao longo do trabalho, discutiram-se os fundamentos teóricos, os trabalhos relacionados e as metodologias adotadas. A partir disso, nas seções deste capítulo, são destacadas as contribuições do estudo para o campo da educação em computação, e sugestões para investigações futuras que possam aprofundar os conhecimentos na área.

### 6.1 Principais contribuições

A presente pesquisa contribui para o campo da computação, especialmente no contexto da educação em Computação para o Ensino Fundamental, com a síntese de uma pesquisa bibliográfica sobre trabalhos relacionados ao ensino de Pensamento Computacional e ao uso do OctoStudio para o ensino de Computação alinhado a outras disciplinas e habilidades.

Através deste estudo, pôde-se perceber o potencial da aplicação do OctoStudio, observou-se o estímulo ao Pensamento Computacional dos alunos, favorecendo habilidades como abstração, decomposição e criação de algoritmos. Além de elevar o engajamento dos estudantes, graças à natureza lúdica e acessível da ferramenta, ela promoveu maior participação nas atividades propostas. A facilidade de acesso ao OctoStudio, por ser gratuito, intuitivo e compatível com dispositivos móveis, representa um diferencial importante para escolas com infraestrutura limitada.

Adicionalmente, este estudo propôs uma sequência didática estruturada alinhada às diretrizes da BNCC, que pode ser adaptada por outros docentes interessados em integrar o Pensamento Computacional de forma transversal no currículo por meio de recursos criativos.

Como principal desafio identificado durante a aplicação do curso foi a dificuldade em manter o foco dos alunos durante as aulas. Por existirem outros aplicativos instalados nos *smartphones*, alguns alunos tinham que ser chamados durante as aulas porque estavam utilizando outro aplicativo e acabavam se distraíndo. Além disso, limitações nos comandos de entrada e de manipulação de variáveis do OctoStudio dificultou algumas possibilidades de interação, como a falta de um evento “espere resposta” ou manipulação de *strings*.

## 6.2 Trabalhos futuros

A partir dos resultados obtidos neste estudo, torna-se possível vislumbrar diversas possibilidades para pesquisas futuras relacionadas ao uso de ferramentas tecnológicas no ensino de Pensamento Computacional. Recomenda-se, por exemplo, a realização de investigações em diferentes etapas da educação básica, a fim de avaliar a eficácia do OctoStudio em contextos variados e com faixas etárias distintas.

Sugere-se a ampliação do número de participantes e a aplicação em escolas com diferentes perfis socioeconômicos, o que permitiria uma análise mais abrangente sobre a adaptabilidade da ferramenta.

Outra vertente de pesquisa promissora consiste na comparação entre o OctoStudio e outras plataformas educacionais voltadas à programação e à criatividade digital disponíveis para dispositivos móveis, buscando identificar os pontos fortes e as limitações de cada abordagem.

Também é relevante explorar estratégias de formação docente mais aprofundadas, com o objetivo de tornar os professores facilitadores de experiências inovadoras com tecnologias educacionais, buscando integrar a tecnologia em suas disciplinas.

Por fim, uma possibilidade de trabalho futuro é investigar o impacto de projetos interdisciplinares que integrem o Pensamento Computacional a áreas como artes, ciências e matemática, fortalecendo a aprendizagem significativa e o desenvolvimento de competências do Século XXI.

## REFERÊNCIAS

BATISTA, Esteic Janaina Santos; JÚNIOR, Amaury Antônio de Castro; LIMA, Anderson Correa de; RIEDNER, Daiani Damm Tonetto; ARGUELO, Miriam Brum; BARBOSA, Ellen Regina Romero; SANDIM, Hercules da Costa. Computação criativa com octostudio no ensino fundamental. In: SBC. **Simpósio Brasileiro de Computação na Educação Básica (SBC-EB)**. [S.l.], 2025. p. 69–73.

BRACKMANN, Christian Puhlmann. **Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica**. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul- UFRGS, Rio Grande do Sul- Brasil, 2017.

BRASIL. Lei nº 15.100, de 13 de janeiro de 2025. **Diário Oficial [da] República Federativa do Brasil**, Brasília, DF, 2025. Disponível em: <https://www.in.gov.br/web/dou/-/lei-n-15.100-de-13-de-janeiro-de-2025-606772935>. Acesso em: 14 de jan. de 2025.

Fonseca, J. J. S. **Metodologia da pesquisa científica**. [S.l.], 2002. Apostila.

Gerhardt, Tatiana Engel; Silveira, Denise Tolfo. **Métodos de pesquisa**. [S.l.]: Porto Alegre: Editora da UFRGS, 2009.

GOOGLE. **Google Acadêmico**. 2004. Página inicial. Disponível em: <https://scholar.google.com/?hl=pt-BR>. Acesso em: 1º jul. de 2025.

IBGE, Instituto Brasileiro de Geografia e Estatística. **Pesquisa Nacional por Amostra de Domicílios Contínua 2021**. Brasil: IBGE, 2021. Disponível em: [cidades.ibge.gov.br/brasil/panorama](http://cidades.ibge.gov.br/brasil/panorama). Acesso em: 11 jul. 2025.

International Society for Technology in Education e Computer Science Teachers Association. **Computational Thinking in K-12 Education: Teacher Resources**. National Science Foundation, 2011. Disponível em: [https://cdn.iste.org/www-root/2020-10/ISTE\\_CT\\_Teacher\\_Resources\\_2ed.pdf](https://cdn.iste.org/www-root/2020-10/ISTE_CT_Teacher_Resources_2ed.pdf).

Lifelong Kindergarten. **OctoStudio**. 2025. Disponível em: <https://octostudio.org/>. Acesso em: 30 de junho de 2025.

Massachusetts Institute of Technology. **OctoStudio**. 2025. Versão 1.0.34. Biblioteca de Projetos.

Ministério da Educação. **Anexo ao Parecer CNE/CEB nº 2/2022**. 2022. Assunto: Normas sobre Computação na Educação Básica - Anexo. Disponível em: <https://www.gov.br/mec/pt-br/escolas-conectadas/BNCCComputaoCompletoDiagramado.pdf>. Acesso em: 10 jul. 2025.

Ministério da Educação. **Parecer CNE/CEB nº 2/2022, de 17 de fevereiro de 2022**. 2022. Brasília, DF: MEC. Assunto: Normas sobre Computação na Educação Básica – Complemento à Base Nacional Comum Curricular. Disponível em: <https://portal.mec.gov.br/programa-escola-acessivel/30000-uncategorised/61911-base-nacional-comum-curricular-bncc-2>. Acesso em: 30 jun. 2025.

NADAL, Clara Pérez. **La combinació de llenguatges de programació amb llenguatge artístic com a recurs didàctic per potenciar la creativitat i la competència digital**. 2024.

OCTOSTUDIO.ORG. **Cartões de programação do OctoStudio**. [S.l.]: Lifelong Kindergarten, 2024.

OCTOSTUDIO.ORG. **Guia de Referência do OctoStudio**. [S.l.]: Lifelong Kindergarten, 2024.

PAPERT, Seymour. **Microworlds: Incubators for knowledge. Mindstorms: Children, Computers, And Powerful Ideas**. [S.l.]: Basic Books, 1980.

PINHO, Robério Girão. **Estimulando o pensamento computacional no ensino médio utilizando o software Scratch**. Russas, 2024.

RAABE, A.; ZORZO, A.F.; BLIKSTEIN, P. **Computação na Educação Básica: Fundamentos e Experiências**. Penso Editora, 2020. (Tecnologia e Inovação na Educação Brasileira). ISBN 9786581334048. Disponível em: <https://books.google.com.br/books?id=EBvLDwAAQBAJ>.

RESNICK, Mitchel. **Learn to Code, Code to Learn**. EdSurge, 2013. Disponível em: <https://www.edsurge.com/news/2013-05-08-learn-to-code-code-to-learn>. Acesso em: 08 de julho de 2025.

Ribeiro, Leila; CASTRO, Alberto; FRÖHLICH, Antônio Augusto; FERRAZ, Carlos Andre Guimaraes; FERREIRA, Carlos Eduardo; SEREY, Dalton; CORDEIRO, Daniel de Angelis; AIRES, José; BIGOLIN, Nara; CAVALHEIRO, Simone. **Diretrizes da Sociedade Brasileira de Computação para o Ensino de Computação na Educação Básica**. 2019. Sociedade Brasileira de Computação, Relatório Técnico, n. 001.

Rusk, Natalie; JAIN, Rupal; MARTIN, Caitlin K.; ROQUE, Ricarose; FREITAS, João Adriano; MOLAODI, Linford. Honoring practices of community-based educators: lessons learned from the collaborative design of a creative mobile app. **Learning, Media and Technology**, Routledge, v. 49, n. 5, p. 794–810, 2024. Disponível em: <https://doi.org/10.1080/17439884.2024.2435202>.

SCRATCH. **Sobre o Scratch**. 2023. Disponível em: <https://scratch.mit.edu/about>. Acesso em: 01 de julho 2025.

ScratchJr. **ScratchJr**. 2014. Disponível em: <https://www.scratchjr.org/>. Acesso em: 19 de fevereiro 2026.

SHIMASAKI, Rodrigo; PRADO, Maria Elisabette Brisola Brito. O ensino da programação e o desenvolvimento do pensamento lógico: uma revisão sistemática de literatura. **Rev. Ensino Educ. Ciênc. Humanas**, Editora e Distribuidora Educacional, v. 22, n. 2, p. 197–205, jun. 2021.

SOL. **Biblioteca Digital da Sociedade Brasileira de Computação - SBCOpenLib (SOL)**. 2018. Página inicial. Disponível em: <https://sol.sbc.org.br/index.php/indice/index>. Acesso em: 01 de julho 2025.

Todos Pela Educação. **Anuário Brasileiro da Educação Básica: Capítulo 12 -Infraestrutura Escolar**. Brasil: Moderna, 2024. Disponível em: <https://anuario.todospelaeducacao.org.br/capitulo-12-infraestrutura.html>. Acesso em: 11 de julho de 2025.

WANGENHEIM, Christiane Gresse von; PETRI, Giani; ZIBETTI, André Wüst; HAUCK, Jean; PACHECO, Fernando Santana; FILHO, Raul. **dTECT: Um Modelo para a Avaliação de Unidades Instrucionais para o Ensino de Computação na Educação Básica**. 2017.

WING, Jeannette M. Computational thinking. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 49, n. 3, p. 33–35, mar. 2006. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/1118178.1118215>.

XISTO, Thaís. **Playful Occupations: Mobile Creative Coding for Critical Consciousness**. [S.l.]: Massachusetts Institute of Technology, 2023.

**APÊNDICE A – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO**

## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Prezado senhor(a), estamos solicitando a sua autorização para permitir que o menor pelo qual você é responsável participe da pesquisa proposta. A participação é voluntária, e você tem o direito absoluto de recusar sua autorização, em qualquer fase da pesquisa.

### 1) Procedimento e Objetivo

Durante o processo, o participante será convidado(a) a responder a um formulário que abordará questões relacionadas às suas experiências em ações de extensão no ensino fundamental. Neste estudo, temos como objetivo avaliar como as ações de extensão no ensino fundamental impactam no desenvolvimento de habilidades relacionadas à computação.

### 2) Tratamento de possíveis riscos e desconfortos

Reconhecemos que esse processo pode despertar desconforto ao recordar experiências passadas e emoções desafiadoras. Para garantir a privacidade do participante, todas as respostas serão mantidas estritamente confidenciais.

### 3) Benefícios e Custos

Este estudo contribuirá com resultados importantes para a pesquisa de um modo geral nas áreas de Engenharia de Software e Ciência da Computação.

Você não terá nenhum gasto ou obrigação com a sua participação no estudo, sendo necessário apenas um aparelho celular ou tablet para a participação. Também não receberá qualquer espécie de reembolso ou gratificação devido à participação na pesquisa.

Caso não seja possível a disponibilização do aparelho, o menor ainda poderá participar, dividindo o uso de um aparelho com algum(ns) colega(s).

### 4) Confidencialidade da Pesquisa

Os dados fornecidos pelo participante serão tratados com a máxima confidencialidade e não serão divulgados de forma que possam identificá-lo(a). As informações coletadas serão utilizadas apenas em congressos ou publicações científicas.

### 5) Organização da Pesquisa

Os pesquisadores responsáveis pelo estudo poderão fornecer qualquer esclarecimento sobre o mesmo, assim como tirar dúvidas, bastando entrar em contato pelos seguintes e-mails:

Pesquisador: Mathias Augusto de Melo - mathias\_melo@alu.ufc.br - UFC-RUSSAS

Orientador: Markos Oliveira Freitas - markos\_freitas@yahoo.com.br - UFC-RUSSAS

### 6) Declaração de Consentimento

Eu, \_\_\_\_\_,

responsável legal por \_\_\_\_\_,

declaro ter sido informado(a) e concordo com a participação do(a) meu filho (a) como participante na pesquisa.

Data: \_\_\_\_\_ Local: \_\_\_\_\_

**APÊNDICE B – TERMO DE ASSENTIMENTO**



UNIVERSIDADE  
FEDERAL DO CEARÁ

## TERMO DE ASSENTIMENTO

Olá! Me chamo Mathias Augusto de Melo, e estou concluindo o Curso de Ciência da Computação pela Universidade de Federal do Ceará.

Você está sendo convidado(a) a participar de uma pesquisa de trabalho de conclusão de curso chamada “O uso do OctoStudio para o ensino de Pensamento Computacional no ensino fundamental”.

Sua participação é completamente voluntária, e você não deve aceitar participar se não desejar. Por favor, leia atentamente as informações a seguir e sinta-se à vontade para fazer qualquer pergunta caso tenha dúvidas.

### 1) Procedimento e Objetivo

Durante o processo, o participante será convidado(a) a responder a um formulário que abordará questões relacionadas às suas experiências em ações de extensão no ensino fundamental. Neste estudo, temos como objetivo avaliar como as ações de extensão no ensino fundamental impactam no desenvolvimento de habilidades relacionadas à computação.

### 2) Tratamento de possíveis riscos e desconfortos

Reconhecemos que esse processo pode despertar desconforto ao recordar experiências passadas e emoções desafiadoras. Para garantir a privacidade do participante, todas as respostas serão mantidas estritamente confidenciais.

### 3) Benefícios e Custos

Este estudo contribuirá com resultados importantes para a pesquisa de um modo geral nas áreas de Engenharia de Software e Ciência da Computação.

Você não terá nenhum gasto ou obrigação com a sua participação no estudo, sendo necessário apenas um aparelho celular ou tablet para a participação. Também não receberá qualquer espécie de reembolso ou gratificação devido à participação na pesquisa.

Caso não seja possível a disponibilização do aparelho, o menor ainda poderá participar, dividindo o uso de um aparelho com algum(ns) colega(s).

### 4) Confidencialidade da Pesquisa

Para participar deste estudo, você será informado sobre qualquer aspecto que desejar e o responsável por você deverá autorizar, assinando um termo. Mesmo que seu responsável autorize sua participação, você ainda tem o direito de recusar, estando livre para participar ou não. Lembre-se de que você e/ou seu responsável poderão deixar de participar a qualquer momento, sem nenhum problema.

### 5) Organização da Pesquisa

Os pesquisadores responsáveis pelo estudo poderão fornecer qualquer esclarecimento sobre o mesmo, assim como tirar dúvidas, bastando entrar em contato pelos seguintes e-mails: **Pesquisador:** Mathias Augusto de Melo - mathias\_melo@alu.ufc.br - UFC-RUSSAS **Orientador:** Markos Oliveira Freitas - markos\_freitas@yahoo.com.br - UFC-RUSSAS

### 6) Declaração de Consentimento

Li ou alguém leu para mim as informações contidas neste documento antes de assinar este termo de consentimento. Declaro que concordo em participar desse estudo.

Participante: \_\_\_\_\_

Data: \_\_\_\_\_ Local: \_\_\_\_\_

**APÊNDICE C – SLIDES DAS AULAS**

Introdução a programação com OctoStudio

# AULA 2

Revisão e  
Comandos de controle



## Quando iniciar?

Bloco	Descrição	Dicas e Opções
	Inicia o roteiro quando você toca o botão Play abaixo do Palco	
	Inicia o roteiro quando você toca no ator selecionado ou outra opção	Escolha qual tocar: ator cenário qualquer

## Movimento

Bloco	Descrição	Dicas e Opções
	Diz a um ator quando ir para o Palco	Escolha um destino indicando . Você pode arrastar o ator na janela, digitar números, ou utilizar aleatoriamente

## Palavras e Sons

Bloco	Descrição	Dicas e Opções
	Faz o ator falar em voz alta	Para alterar a pronúncia do idioma, altere o idioma nas configurações do OctoStudio.
	Exibe palavras em um balão de fala	Digite por quantos segundos você deseja que as palavras apareçam.



## Controle

Bloco	Descrição	Dicas e Opções
	Verifica uma vez se a condição é verdadeira. Neste caso, executa os blocos de dentro.	Para adicionar um "se" ao bloco, clique em  uma vez. Para o seguimento "e se", clique  múltiplas vezes. Você pode inserir um número, variável ou outra informação como: 
	Aguarda um número específico de segundos antes de continuar	

## Mais Blocos

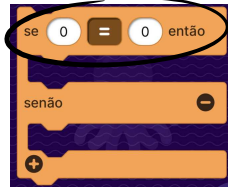
Bloco	Descrição	Dicas e Opções
	Define a variável selecionada para um número específico	Para criar uma nova variável, toque em . Atualmente, os nomes das variáveis não podem ser excluídos.
	Mostra ou esconde o monitor variável no Palco	Para esconder a variável, defina . As variáveis são exibidas no Palco por padrão.
	Retorna ao valor numérico atual de uma variável	Você pode encaixar este bloco em um espaço vazio dentro de outro bloco

# Mais Blocos

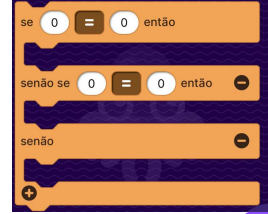
Bloco	Descrição	Dicas e Opções
 1 to 10	Escolhe um número aleatório dentro do intervalo especificado	
	Executa uma determinada operação matemática em dois valores.	Escolha adição <b>+</b> , subtração <b>-</b> , multiplicação <b>x</b> , ou divisão <b>÷</b> .

Se... então

Condição



Muitas opções!



## Exercícios

## Exercícios

## Exercícios

1



2

num1  
76

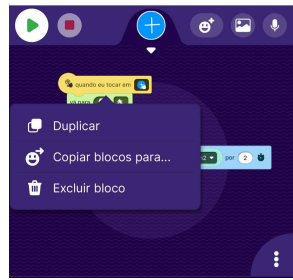


num2  
30

106

## Atividade de casa

## Dica



Introdução a programação com OctoStudio

# AULA 3

Revisão e Comandos de Repetição



## Quando iniciar?

Bloco	Descrição	Dicas e Opções
	Quando o botão <i>play</i> é tocado, executa o script abaixo	
	Inicia o ator quando você agita o celular (ou tablet)	Escolha o nível de agitação necessário para iniciar: baixo  médio  alto
	Inicia o roteiro quando você toca no ator selecionado ou em outra opção	Escolha tocar o ator  cenário  qualquer
	Inicia o roteiro quando o ator atinge a borda selecionada do Palco	Só começa se o ator atingir a borda de <b>dentro</b> do Palco (não quando passar para o outro lado).

## Movimento

Bloco	Descrição	Dicas e Opções
	Mova o ator em uma direção em uma velocidade específica	Escolha direção com a seta, defina velocidade  lenta  média  rápida  instantânea
	Gira o ator em um certo número de graus	Digite um ângulo de 1 a 360. Defina a direção para o sentido:  horário  anti-horário  aleatório
	Diz a um ator quando ir para o Palco	Escolha um destino indicando . Você pode arrastar o ator na janela, digitar números, ou utilizar aleatoriamente


## Movimento

Bloco	Descrição	Dicas e Opções
	Define a direção em que um ator está indo	Este bloco irá virar o ator. Utilize o bloco de girar para a rotação completa.
	Avança na direção atual	






## Cena

Bloco	Descrição	Dicas e Opções
	Define o tamanho do ator para uma % especificada do tamanho original	
	Altera o tamanho do ator por quantidade especificada	






# Palavras e Sons

Bloco	Descrição	Dicas e Opções
	Toca um som gravado ou da biblioteca	Clique para escolher ou gravar som. Escolha "até o fim" ► se você quiser que este som termine de tocar antes de acionar os próximos blocos. Escolha 'e continue' ► se você quiser que toque ao mesmo tempo que o bloco seguinte.




# Mais Blocos

Bloco	Descrição	Dicas e Opções
	Define a variável selecionada para um número específico	Para criar uma nova variável, toque em +. Atualmente, os nomes das variáveis não podem ser excluídos.
	Retorna ao valor numérico atual de uma variável	Você pode encaixar este bloco em um espaço vazio dentro de outro bloco
	Escolhe um número aleatório dentro do intervalo especificado	
	Mostra ou esconde o monitor variável no Palco	Para esconder a variável, defina  . As variáveis são exibidas no Palco por padrão.

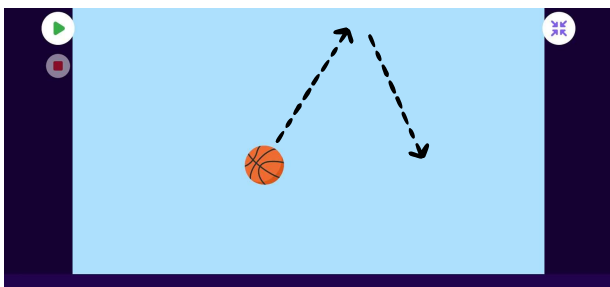
# Controle

Bloco	Descrição	Dicas e Opções
	Aguarda um número específico de segundos antes de continuar	
	Envia uma mensagem que todos os atores recebem	Exemplos:  
	Inicia um roteiro quando a mensagem selecionada é recebida	

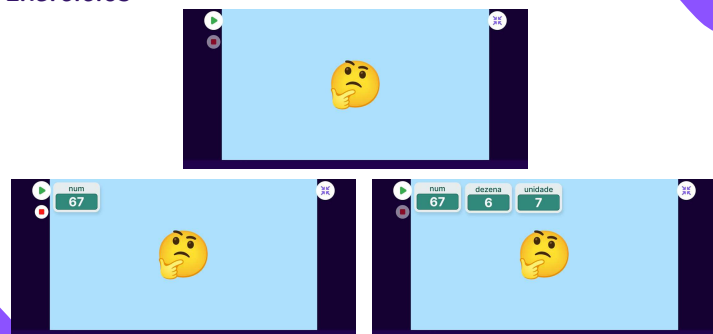
# Controle

Bloco	Descrição	Dicas e Opções
	Executa o script dentro deste bloco um número específico de vezes	Encaixe os blocos dentro do espaço vazio do bloco de <b>repetir</b> ou <b>sempre</b>
	Executa o roteiro repetidamente até que o projeto pare	Exemplo: 

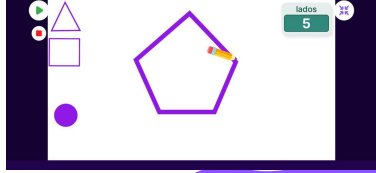
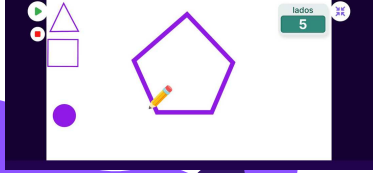
## Exercícios



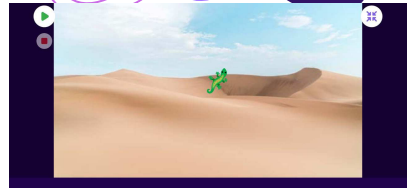
## Exercícios



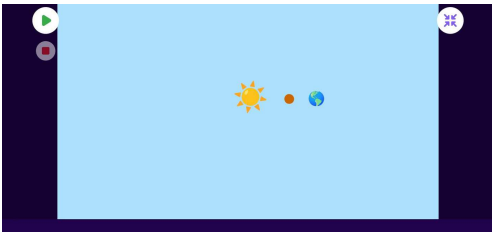
Exercícios



Exercícios



Exercícios



**ANEXO A – CONTEÚDOS E HABILIDADES DE COMPUTAÇÃO PARA O 6º ANO -  
EIXO DE PENSAMENTO COMPUTACIONAL**

## COMPUTAÇÃO - 6º ANO

EIXO	OBJETO DE CONHECIMENTO		HABILIDADE	
PENSAMENTO COMPUTACIONAL	Programação	Tipos de dados	(EF06CO01) Classificar informações, agrupando-as em coleções (conjuntos) e associando cada coleção a um 'tipo de dados'	
		Linguagem de programação	(EF06CO02) Elaborar algoritmos que envolvam instruções sequenciais, de repetição e de seleção usando uma linguagem de programação.	
	Estratégias de solução de problemas	Decomposição	Construir e analisar soluções computacionais de problemas de diferentes áreas do conhecimento, de forma individual ou colaborativa, selecionando as estruturas de dados adequadas (registros, matrizes, listas e grafos), aperfeiçoando e articulando saberes escolares.	(EF06CO03) Descrever com precisão a solução de um problema, construindo o programa que implementa a solução descrita.
				(EF06CO04) Construir soluções de problemas usando a técnica de decomposição e automatizar tais soluções usando uma linguagem de programação.
		Generalização	Empregar diferentes estratégias da Computação (decomposição, generalização e reuso) para construir a solução de problemas.	(EF06CO05) Identificar os recursos ou insumos necessários (entradas) para a resolução de problemas, bem como os resultados esperados (saídas), determinando os respectivos tipos de dados, e estabelecendo a definição de problema como uma relação entre entrada e saída.
				(EF06CO06) Comparar diferentes casos particulares (instâncias) de um mesmo problema, identificando as semelhanças e diferenças entre eles, e criar um algoritmo para resolver todos, fazendo uso de variáveis (parâmetros) para permitir o tratamento de todos os casos de forma genérica.

	EXPLICAÇÃO DA HABILIDADE	EXEMPLOS
	As informações são armazenadas de diferentes maneiras, dependendo do tipo de dado que ela representa. Basicamente existem três tipos primitivos de dados: inteiros, real e string.	Encontrar um Ás em um baralho, precisa-se de um baralho (lista de cartas) e, o resultado é uma carta; para calcular a média das provas dos alunos de uma turma, precisa-se da lista de provas dos alunos, e o resultado é um número.
	Existem diferentes linguagens de programação que podem ser usadas para descrever algoritmos em diferentes níveis de abstração, como linguagens visuais, orientadas a objetos, funcionais, entre outras. Uma ou mais linguagens podem ser escolhidas para serem adotadas.	Calcular a média de notas de uma turma em uma dada disciplina e informar se o resultado está acima da média do colégio.
	É importante que se consiga expressar a solução do problema (algoritmo) em português, compreendendo que o programa é apenas uma descrição deste algoritmo em uma linguagem de programação. O aluno precisa entender que o mais importante é a construção do algoritmo. Notem que a ideia aqui não é apenas descrever as linhas de código em português, mas sim descrever em um alto nível de abstração como o problema é resolvido.	Desenvolver um programa que: (1) "Se o ponteiro do mouse tocar no animal então o animal andará 10 passos, 10 vezes seguidas." (2) "Dada uma pilha de cartas, se a pilha estiver vazia, dizer que não há ás; se a primeira carta for um ás, dizer que há ás na pilha, senão, remover a primeira carta e verificar se há ás no resto da pilha."
	Decomposição é uma das principais técnicas de resolução de problemas, onde um problema é dividido em subproblemas, os quais são resolvidos independentemente, e cujas soluções são combinadas para construir a solução do problema original. Algumas vantagens da decomposição são: permitir uma melhor organização e visualização do problema e da solução; facilitar o trabalho em grupo; permitir que possamos reutilizar as soluções dos subproblemas em outros problemas.	Decompor o problema de desenhar imagens em subproblemas de desenhar formas básicas, compondo as subsoluções por meio de operações sobre imagens (sobrepor, posicionar ao lado etc.). Decompor o problema de desenhar uma casa em subproblemas de desenhar polígonos regulares (retângulos, quadrados, triângulos), compondo essas formas com as operações sobre imagens (rotação, sobreposição etc.).
	Definir problemas é uma habilidade muito importante, pois é o primeiro passo da solução. A definição de um problema se dá identificando quais são os tipos de entradas necessárias (insumos/informações) e qual o tipo da saída. Como a solução (algoritmo) deve ser genérica, se define um problema em termos dos tipos das entradas e saída. O objetivo aqui NÃO é propor soluções de problemas, e sim definir o que é necessário para resolvê-los e qual será o resultado esperado.	Encontrar um Ás em um baralho, precisa-se de um baralho (lista de cartas) e, o resultado é uma carta; para calcular a média das provas dos alunos de uma turma, precisa-se da lista de provas dos alunos, e o resultado é um número.
	Idealmente, um algoritmo é uma solução genérica: ele resolve várias instâncias de um problema. Por exemplo, um algoritmo que calcula a média aritmética de 2 números resolve este problema para qualquer par de números (que são as instâncias do problema). Para descrever um algoritmo de forma genérica, é necessário dar nomes às entradas do algoritmo. Esses nomes são chamados de variáveis ou parâmetros do algoritmo.	Comparar diferentes instâncias do problema de calcular a área de um retângulo, identificando que o que varia entre elas são as medidas da base e da altura e, por fim, criar um algoritmo para calcular a área de qualquer retângulo.

## ANEXO B – TABELA DE QUESTÕES DO DETECT

No.	Descrição do Item	Formato das respostas
<b>Qualidade da Unidade Instrucional</b>		
1	A oficina foi:	(1) Muito Divertida (2) Divertida (3) Chata (4) Muito chata
2	O tempo das aulas passou:	(1) Muito rápido (2) Rápido (3) Devagar (4) Muito devagar
3	A oficina foi:	(1) Excelente (2) Boa (3) Regular (4) Ruim
<b>Experiência de Computação</b>		
4	Vou mostrar meu programa de computador para outras pessoas:	(1) Sim (2) Não
5	Quero aprender mais sobre como fazer programas de computador:	(1) Sim (2) Não
6	Fazer um programa de computador é:	(1) Muito Divertido (2) Divertido (3) Chato (4) Muito chato
7	Eu gosto de fazer programas de computador:	(1) Sim (2) Não
8	A computação é útil no dia a dia:	(1) Sim (2) Não
9	Você já pensou em trabalhar com computação?	(1) Sim (2) Não
<b>Percepção da Aprendizagem</b>		
10	A oficina foi:	(1) Muito fácil (2) Fácil (3) Difícil (4) Muito Difícil
11	Eu consigo fazer programas de computador:	(1) Sim (2) Não
12	Consigo explicar para um amigo/amiga como fazer um programa de computador:	(1) Sim (2) Não
13	Fazer um programa de computador é:	(1) Muito fácil (2) Fácil (3) Difícil (4) Muito Difícil

**ANEXO C – GUIA DE REFERÊNCIA DO OCTOSTUDIO**

# octostudio

## Guia de Referência

Revisado em 10 Julho, 2025



### Introdução

O OctoStudio é um aplicativo gratuito de programação que possibilita criar animações, jogos e outros projetos interativos utilizando fotos, desenhos, e sons do mundo ao seu redor. Use os blocos de programação para dar vida às suas ideias. O OctoStudio é desenvolvido pelo grupo Lifelong Kindergarten, o grupo no MIT Media Lab que criou o ambiente de programação Scratch.



**Uma ótima forma de aprender a criar com o OctoStudio é explorando e brincando!**

Este guia de referência inclui uma detalhada descrição dos blocos de programação, ícones, configurações, e outras funcionalidades no OctoStudio. Seu propósito é ser referência ao invés de instrução. Se você procura recursos pedagógicos e guias do educador, recomendamos que visite a página [Recursos de Aprendizagem para o OctoStudio](#) no website [octostudio.org](#).

Você pode baixar o OctoStudio gratuitamente nas lojas de aplicativo:



Para mais detalhes, veja **Requisitos de Sistema** ao final deste guia.

Se você tem dúvidas, por gentileza visite a seção [Suporte](#) na página do OctoStudio ou envie uma mensagem de e-mail para [help@octostudio.org](mailto:help@octostudio.org).

### Introdução

2

### Interface

Página Inicial	3
Visão Geral do Editor de Projeto	4
Botões de Edição do Projeto	6
Atores	7
Cenários	8
Palco	8
Sons	9
Menu Compartilhar	10

### Blocos de Programação

Categorias de Blocos	11
Opções de Velocidade	11
Quando começar?	12
Movimento	13
Palavras e Sons	14
Cenário	15
Cores e Luz	16
Controle	17
Mais Blocos	18

### Instalação e Configurações

Configurações	21
Requisitos do Sistema	22
OctoStudio em Chromebooks	22

Este guia de referência foi criado sob a licença Creative Commons Atribuição-Compartilhual.

Nós pedimos que você inclua este crédito:

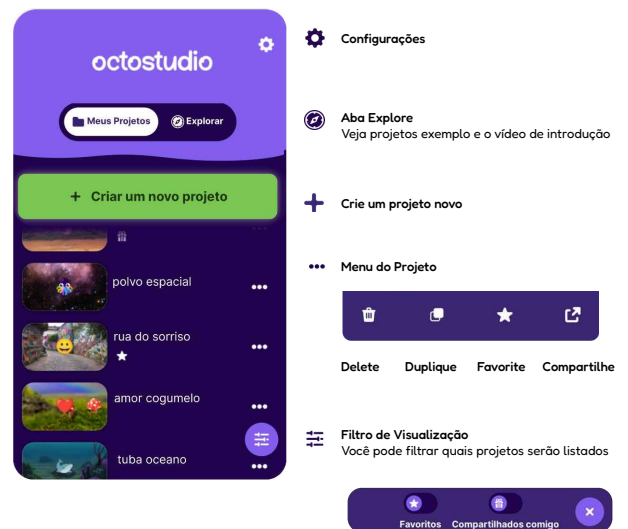
O OctoStudio foi criado pelo Grupo Lifelong Kindergarten no MIT Media Lab. Para saber mais, visite [octostudio.org](#).



### Interface

#### Página Inicial

Todos os seus projetos são listados na página inicial. Você também pode ir na aba Explore para experimentar projetos exemplo e assistir ao vídeo de introdução. Para acessar Configurações, use o ícone no topo a direita.



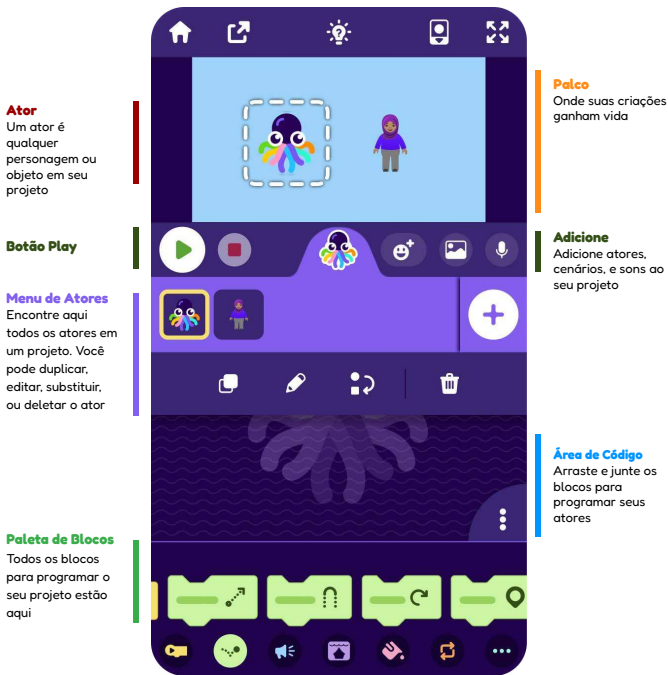
**Favoritos:** Liste apenas projetos que você favoritou com uma estrela.

**Compartilhados comigo:** Liste apenas projetos que foram compartilhados com você.

## Visão Geral do Editor de Projeto

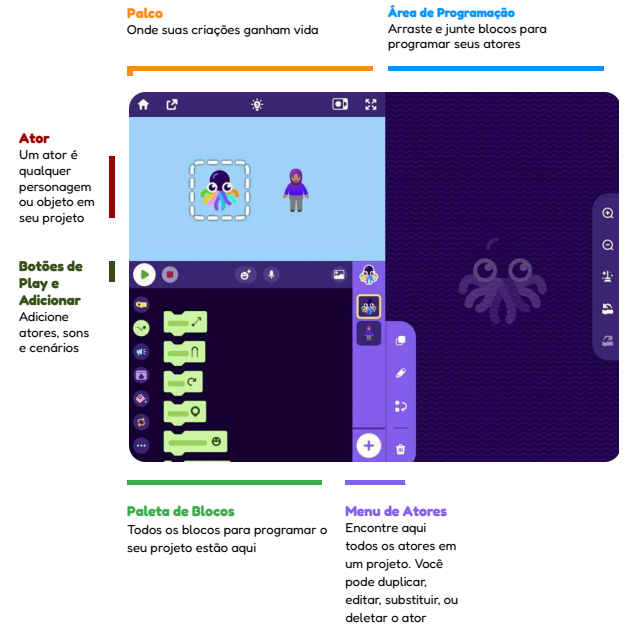
### Modo Retrato

Em celulares e na maioria dos tablets, o editor de projeto aparece em Modo Retrato.

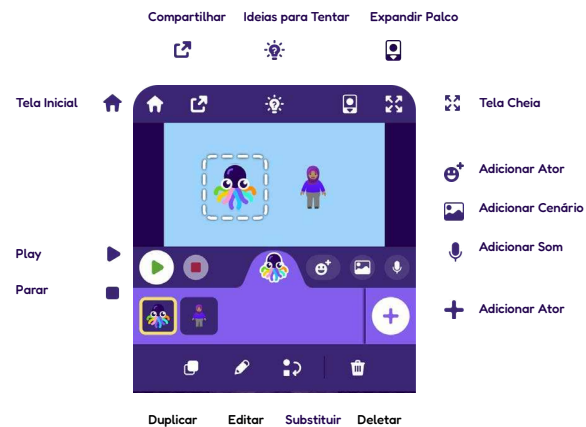


### Modo Paisagem

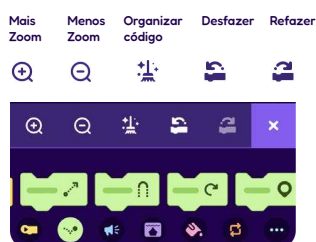
Em Chromebooks e tablets grandes, você pode usar o editor de projeto em Modo Paisagem. Esta interface contém as mesmas opções do Modo Retrato, reorganizadas no espaço disponível.



## Botões de Edição do Projeto



### Opções de Programação



## Atores

Um ator é qualquer personagem, objeto ou imagem ao qual você pode dar vida com blocos de programação.

### Adicione um Ator

Há quatro modos diferentes de adicionar um ator.



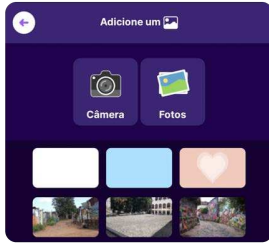
### Editando Foto como Ator



Após você criar um ator, você pode duplicar, editar, substituir, ou deletá-lo utilizando os botões no Menu de Atores do Editor do Projeto. Para editar seu Ator, procure o ícone lápis no Menu de Atores.

## Cenários

Você escolhe um cenário quando cria um projeto. Você também pode adicionar mais cenários no Editor do Projeto.



Há três opções diferentes para adicionar um cenário:



**Tire uma foto**



**Escolha do seu álbum de fotos**



**Biblioteca de Cenários**

Escolha da nossa coleção de cenários, incluindo fotos de nossos parceiros globais.

## Palco



A área da tela onde seu projeto é apresentado se chama **Palco**.

O Palco tem 320 unidades de largura e 180 de altura, formando uma grade x-y. O canto inferior do Palco está na posição x: 0, y: 0.

## Sons

Grave seus próprios sons ou selecione um som da biblioteca para utilizar no seu projeto.



**Escolher um som ou Gravar um som**

**Duração do Som** em segundos

**Selecione para editar**

Arraste as barras laterais para selecionar parte do som. Para recortar, selecione a parte que você quer manter e depois selecione o botão Recortar que ficará visível



**O botão de Regravar** possibilita que você substitua a gravação atual

**Efeitos**

Você pode adicionar efeitos de som a um som. Efeitos podem ser adicionados apenas ao trecho selecionado da gravação



**Mais rápido**

Faça a gravação tocar rápido



**Mais devagar**

Faça a gravação tocar devagar



**Eco**

Adicione um efeito de eco a sua gravação



**Robótico**

Adicione um efeito robótico a sua gravação

## Menu Compartilhar

Você pode compartilhar seu projeto como um vídeo, GIF ou arquivo do OctoStudio a partir do menu **Compartilhar**.

Clique neste ícone no topo do Editor do Projeto para acessar o menu Compartilhar.



**Grave um Vídeo**

Grave um vídeo de seu projeto para compartilhar. Cada vídeo pode ter até 30 segundos de duração.



**Faça um GIF**

Faça um loop animado do seu projeto. GIFs podem ter até 15 segundos.



**Envie o Arquivo do Projeto**

Há diferentes formas de enviar ou compartilhar um arquivo de projeto no OctoStudio.

**Envie o arquivo para alguém:** você pode enviar uma cópia do arquivo usando e-mail, Google Drive, e vários outros aplicativos instalados no seu dispositivo.

A pessoa recebendo o arquivo precisa ter o Octostudio instalado em seu dispositivo para abrir e interagir com o projeto.

**Salve uma cópia em seu dispositivo:** Você pode salvar uma cópia do arquivo do projeto em uma pasta.

**Upload para um ambiente de aprendizagem:** Após você salvar uma cópia do arquivo do projeto em uma pasta no seu dispositivo, você pode então fazer seu upload para o Google Sala de Aula ou outro ambiente de aprendizagem.

## Blocos de Programação

### Categorias de Blocos

Há sete categorias de blocos no OctoStudio.

Nome	Descrição
<b>Quando começar?</b>	Posicione um destes blocos no topo de cada roteiro de código para definir quando iniciar.
<b>Movimento</b>	Faça seus atores se moverem.
<b>Palavras e Sons</b>	Faça seus atores tocarem sons, mostrarem texto, e mais.
<b>Cenário</b>	Faça alterações visuais, como redimensionar seus atores e alterar o cenário.
<b>Cores e Luz</b>	Mude a cor do seu ator e outras configurações de luz.
<b>Controles</b>	Controle o fluxo da sua programação.
<b>Mais Blocos</b>	Inclui blocos diversos, como variáveis, 'tilt' e blocos personalizados.

### Opções de Velocidade

Muitos dos blocos têm um menu que permite selecionar o quão rápido você quer o nível de velocidade da ação do bloco realizada.



Lenta



Média



Rápida



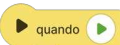
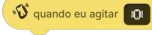




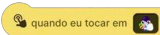




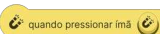
Instantânea





**Exemplo:**






## Quando começar?

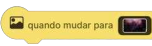
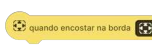
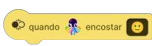
Para fazer algo acontecer no seu projeto, coloque os seus blocos de programação abaixo de qualquer bloco desta categoria.

Bloco	Descrição	Dicas e Opções
	Inicia o roteiro quando você toca no botão Play abaixo do Palco	
	Inicia o roteiro quando você agita o celular ou tablet	Escolha o nível de agitação necessário para iniciar:  Baixa  Média  Alta
	<b>Varição no Chromebook:</b> Inicia quando você toca no botão da tela	Em <b>Chromebooks</b> , o bloco <b>quando agitar</b> tem aparência diferente, como você vê aqui. Ele ativa quando você toca no botão que aparece na tela.
	Inicia o roteiro quando você toca no ator selecionado ou outra opção	Escolha qual tocar:  ator  cenário  qualquer
	Inicia o roteiro quando um ímã é colocado próximo ao celular ou tablet	Mova o ímã sob o celular ou tablet. Tente um ímã forte e teste diferentes locais no dispositivo.
	<b>Varição no Chromebook:</b> Inicia quando você clica no botão na tela	Para aprender mais, assista ao vídeo "Blocos de ímã no OctoStudio" ( <a href="#">OctoStudio Magnet Blocks video</a> ).  A funcionalidade ímã pode não estar disponível em alguns dispositivos Android.











Bloco	Descrição	Dicas e Opções
	Diga a um ator aonde ir no Palco	Toque no ícone de pin no mapa, depois selecione a localização onde você quer que seu ator vá. Para definir a localização, você pode arrastar o ator, digitar em números ou selecionar o aleatório.
	Faz este ator ir para a posição de outro ator	Este ator irá para o <b>ponto de rotação</b> de outro ator. (Você pode mover o ponto de rotação do ator no editor de desenho).
	Define a direção para a qual um ator está olhando	Este bloco irá inverter um ator. Se você quer uma rotação completa, utilize ao invés o bloco <b>girar</b> .
	Mova em frente na direção atual	





## Palavras e Sons

Bloco	Descrição	Dicas e Opções
	Toca um som gravado ou da biblioteca	Clique para escolher ou gravar um som.  Selecione 'até o fim' se você quiser que este som termine de tocar antes de acionar os próximos blocos.  Selecione 'e continue' se você quiser que o som toque ao mesmo tempo em que o bloco seguinte.





Bloco	Descrição	Dicas e Opções
	Inicia o roteiro quando o cenário é trocado para o indicado	Isto é útil para iniciar código quando você muda o cenário ou as cenas.  O roteiro vai iniciar quando o bloco <b>quando mudar para</b> estiver na metade de sua transição.
	Inicia o roteiro quando o ator chega a borda selecionada do Palco	Apenas inicia se o ator alcança a borda por dentro do Palco (não quando o ator passa para o outro lado).
	Inicia o roteiro quando o ator atual toca o ator selecionado	Escolha um ator para detectar. Ao menos dois atores são necessários para este bloco funcionar.

## Movimento

Bloco	Descrição	Dicas e Opções
	Mova o ator em uma direção, em uma velocidade específica	Escolha a direção com a seta, defina a velocidade como:  Lenta  Média  Rápida  Instantânea
	Faz um ator ir para cima e para baixo	Digite um número maior para saltos mais altos. Um número negativo irá fazê-lo saltar ainda mais para baixo.
	Gira o ator em um certo número de graus	Digite no bloco um ângulo de 1 a 360 para fazer o ator girar na direção que você escolher.  Sentido horário  Sentido anti-horário  Aleatório

Bloco	Descrição	Dicas e Opções
	Faz o celular vibrar	Apenas celulares com feedback tátil irão vibrar. Todos os dispositivos irão tocar um som de <i>buzz</i> .
	Faz seu ator falar em voz alta	Para alterar a pronúncia do idioma, altere o idioma nas configurações do OctoStudio.
	Exibe palavras em um balão de diálogo	Digite por quantos segundos você deseja que as palavras apareçam.
	Faz o texto aparecer no Palco como um título ou legenda	Toque no texto dentro do bloco para editar, redimensionar, mudar a cor, ou alterar a localização onde o texto será exibido.  Cada ator pode exibir apenas um bloco de texto por vez no Palco. Para exibir múltiplas caixas de texto ao mesmo tempo, utilize o bloco <b>texto</b> em atores adicionais

## Cenário

Bloco	Descrição	Dicas e Opções
	Define o tamanho do ator para uma % específica do tamanho original	
	Altera o tamanho do ator por quantidade específica	
	Faz o ator desaparecer do Palco	 Para esconder o ator no início de um projeto, escolha a velocidade instantânea. Atores escondidos não interagem com outros atores.

Bloco	Descrição	Dicas e Opções
	Faz o ator aparecer no Palco	Altere a transparência inserindo um número menor do que 100. Em 0 (zero), o ator fica invisível mas ainda interativo.
	Define o cenário para o cenário selecionado backdrop	Escolha um cenário específico ou anterior, próximo ou aleatório. anterior próximo aleatório Você também pode escolher o quão rápido o cenário muda.
	Organiza a posição do ator em frente ou atrás de outro ator	Organize a posição do ator na camada da frente ou atrás de um ator específico ou de todos os atores: todos os atores

## Cores e Luz

Bloco	Descrição	Dicas e Opções
	Cria um brilho colorido ao redor do ator	Escolha a cor do brilho.
	Liga ou desliga a lanterna em celulares ou tablets que têm lanterna embutida	A lanterna retorna ao modo <b>desligada</b> quando você clica em Stop ou Play. Se o seu dispositivo não tem lanterna embutida, você pode ver a palco brilhar ao invés de a lanterna ligar.
	Define a cor específica de um ator ou cenário	Opções: selecione a cor de um ator ou cenário. Para redefinir, selecione sem cor
	Muda a cor do ator ou cenário	Irá retornar para sua cor original a cada 360 unidades.

## Controle

Bloco	Descrição	Dicas e Opções
	Executa o roteiro dentro de um número específico de vezes	Junte blocos dentro do espaço vazio no bloco <b>repita</b> ou <b>sempre</b> .
	Executa o roteiro dentro do bloco repetidamente até que o projeto pare	Exemplo: 
	Verifica se a condição é verdadeira. Se sim, executa os blocos de dentro	Em razão de o bloco <b>se</b> checar apenas uma vez, normalmente funciona melhor dentro de um bloco <b>sempre</b> Você pode incluir um número, variável ou outro valor de repórter posição x Clique no símbolo de mais para adicionar <b>senão</b> a este bloco. Clique novamente para adicionar <b>senão se</b> .
	Espera um número específico de segundos antes de continuar	
	Espera até que um sensor seja detectado, depois continua para acionar o block seguinte	Opções para esperar <b>até tocar</b> , <b>agitar</b> , ou <b>detectar ímã</b> próximo ao celular ou tablet. <b>Variação no Chromebook:</b> A opção <b>agitar</b> é substituída com <b>pressionado</b> para indicar tocar no botão que será exibido na tela. Veja a seção <b>OctoStudio em Chromebooks</b> ao final deste guia para saber mais.

Bloco	Descrição	Dicas e Opções
-------	-----------	----------------

	Interrompe atores ou sons em execução	As opções são para <b>parar este roteiro</b> , <b>parar todos os sons</b> , <b>parar outros roteiros</b> neste ator, ou <b>parar todos os roteiros</b> do projeto.
	Transmite uma mensagem que todos os atores podem receber	Exemplo de <b>transmitir and receber</b> : 
	Inicia um roteiro quando a mensagem selecionada é recebida	




## Mais Blocos

Bloco	Descrição	Dicas e Opções
	Incline seu celular ou tablet para movimentar o ator	Funciona melhor com um bloco <b>sempre</b> . (Por isso ao arrastar este bloco para o roteiro, ele já vem com o <b>sempre</b> ) Opções de inclinação: Horizontal apenas Para Cima e para Baixo apenas Todas as quatro direções <b>Celular ou Tablet:</b> Segure no ângulo desejado antes de clicar em play.
	Informa o ângulo em que seu dispositivo está sendo inclinado, horizontal ou vertical	O valor do ângulo a que este bloco retorna é reduzido para ¼. <b>Chromebook:</b> O controle de inclinação é ativo apenas enquanto o bloco está sendo executado.

Bloco	Descrição	Dicas e Opções
-------	-----------	----------------

	Define a variável selecionada para um valor específico	Para criar uma nova variável, toque em + no menu. Atualmente, os nomes das variáveis não podem ser excluídos.
	Altera a variável selecionada por um determinado valor	
	Mostra ou esconde o monitor variável no Palco	Esconda a variável com <b>desativado</b> Variáveis são exibidas no Palco por padrão.
	Retorna ao valor numérico atual de uma variável	Você pode encaixar este bloco em qualquer espaço vazio arredondado dentro de outro bloco.
	Informa o tamanho ou outras informações sobre o ator	Opções para relatar a direção, tamanho, posição x e y de um ator.
	Escolhe um número aleatório dentro do intervalo especificado	
	Executa uma determinada operação matemática em dois valores	Selecione com o menu: + adição - subtração x multiplicação ÷ divisão

Bloco	Descrição	Dicas e Opções
-------	-----------	----------------

	<p>Envia mensagem via Bluetooth para outros dispositivos próximos usando o OctoStudio</p>	<p>Habilite o Bluetooth para este bloco <b>emita</b> uma mensagem para todos os dispositivos próximos utilizando o OctoStudio.</p>
	<p>Executa o roteiro abaixo quando recebe a mensagem específica emitida via Bluetooth</p>	<p>Existem diferentes mensagens que podem ser emitidas:</p> <p>▲ ● ■ ♥ ☆ ✨ ◆ ◇ ◆</p> <p>Você também pode adicionar um <b>canal</b> dentro do bloco de emissão. Isto é útil quando você tem múltiplos grupos de pessoas trabalhando em diferentes projetos com emissão de sinal em uma mesma localização, para evitar conflito de sinais.</p> <p>O alcance do bloco <b>emita</b> é de aproximadamente 10 metros.</p>
	<p>Crie um bloco personalizado.</p> <p>Nomeie seu bloco e depois encaixe os blocos a seguir para definir o que seu novo bloco irá fazer</p>	<p>Blocos personalizados podem tornar seu código mais organizado. Todos os blocos personalizados aparecerão em 'Mais Blocos' no final da paleta de blocos.</p> <p>Os blocos personalizados são específicos de um ator.</p>

## Requisitos do Sistema

O OctoStudio pode funcionar nos seguintes dispositivos:

Em um **iPhone** ou **iPad**, versão iOS 15 ou superior.  
 Em um celular ou tablet **Android**, versão 8 ou superior.  
 Em **Chromebooks** compatíveis com aplicativos Android.

O OctoStudio ainda não é compatível com outros laptops ou computadores.

## OctoStudio em Chromebooks

OctoStudio em Chromebooks é diferente em certos modos do aplicativo quando utilizado em celulares ou tablets.

### Sensores

A maior parte dos Chromebooks não têm sensores físicos. Então, em Chromebooks, há um modo diferente de interagir com blocos de sensores.



Para blocos **inclin**, você pode interagir utilizando o controle de inclinar que aparece na tela quando o código executa.



Para blocos de **agitar** ou **ímã**, você pode interagir utilizando os botões que aparecem na tela.

### Emita

Chromebooks não podem enviar um sinal, então o bloco **emita** não irá funcionar. No entanto, Chromebooks podem receber um sinal de um celular ou tablet, então **quando o sinal é recebido**, os blocos irão funcionar utilizando o Bluetooth.

### Câmera

Diferentemente de dispositivos móveis, os Chromebooks geralmente possuem apenas uma câmera e frontal.

Para ativar a câmera em um Chromebook, você pode clicar no botão na tela ou pressionar as teclas Espaço ou Enter no teclado.

## Instalação e Configurações

### Configurações

#### Onde encontrar Configurações?

Procure a engrenagem  no canto superior direito da tela inicial.

#### Idiomas

OctoStudio é oferecido em mais de 30 idiomas. Você pode alterar o idioma a qualquer momento nas configurações.

#### Posição da tela

Em tablets grandes você tem a opção de escolher visualizar em modo **Retrato** ou **Paisagem**.



#### Acessibilidade

Para utilizar o OctoStudio com um **leitor de tela**, ligue o leitor de tela incluído no seu dispositivo (**Voiceover** no iOS e **TalkBack** em Android).

Em Configurações do OctoStudio, você também pode ativar a **Sonificação**. Isto fará um som tocar para cada bloco de programação com característica visual. (Por exemplo, você pode escutar um efeito de som quando você clicar em um bloco de pular ou brilhar).

#### Opções de Interface

Se você ativar **Mostrar toques na tela**, você verá um círculo que mostra a localização de seus dedos onde quer que você toque na tela. Isto pode ser útil quando estiver demonstrando como utilizar o OctoStudio.

#### Iniciando

**Dicas iniciais** são ícones animados de mão que mostram os primeiros passos para usar o OctoStudio. Você pode ligar esta opção a qualquer momento.

#### Mais informações

Para mais informações, ideias, e apoio técnico, por gentileza veja a nossa página [Suporte](https://octostudio.org) em [octostudio.org](https://octostudio.org).

Um modo alternativo de adicionar fotos ao seu projeto é tirar uma foto com um celular ou tablet, e enviá-la para o Chromebook.

#### Opções para salvar os arquivos

Você pode salvar uma cópia do projeto em uma pasta de seu dispositivo ou diretamente no **Google Drive**. Então você pode fazer o upload do arquivo para o **Google Sala de Aula** ou outro ambiente virtual de aprendizagem.



#### Dicas de touchpad no Chromebook

Para **rolar**, use dois dedos no touchpad.

Para **mais zoom**: Coloque dois dedos no touchpad e lentamente mova-os para diferentes direções.

Para **menos zoom**: Coloque dois dedos no touchpad e pressione-os juntos.

Para **arrastar**, toque para baixo no touchpad e mova com um dedo. Ou, você pode usar um dedo para pressionar para baixo e um dedo diferente para mover.

Se você tem dúvidas, por favor visite a seção [Suporte](https://octostudio.org) no site do OctoStudio ou envie um e-mail para [help@octostudio.org](mailto:help@octostudio.org)

**ANEXO D – CARTÕES DE PROGRAMAÇÃO DO OCTOSTUDIO**

# Cartões de Programação



# octostudio

octostudio.org

Desenvolvido pelo grupo  
**Lifelong Kindergarten,**  
MIT Media Lab.  
Creative Commons  
CC BY-NC-SA 4.0

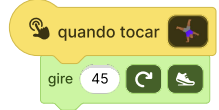
## Gire

Faça seu ator girar

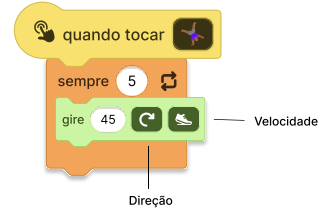


octostudio  
octostudio.org

## Tente este código

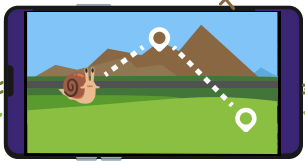


## Explore mais



## Vá para

Vá de um lugar para outro



octostudio  
octostudio.org

## Tente este código



## Explore mais



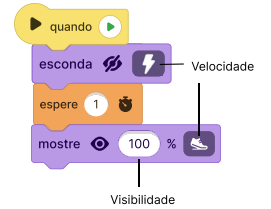
## Esconda e Mostre

Faça algo sumir e aparecer

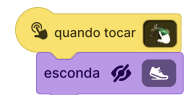


octostudio  
octostudio.org

## Tente este código



## Explore mais



## Aumente e diminua

Faça algo aumentar e diminuir



octostudio  
www.octostudio.org

## Tente este código

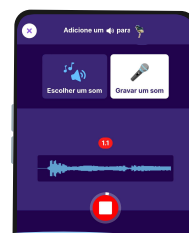


## Explore mais



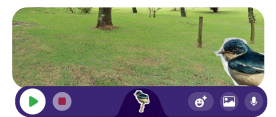
## Adicione som

Grave sua voz ou adicione outros sons



octostudio  
octostudio.org

## Adicione um som



Grave um som

## Tente este código

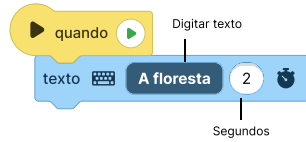


## Adicione texto

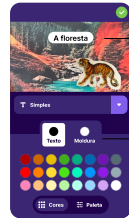
Escreva algo em seu projeto



## Tente este código



## Explore mais



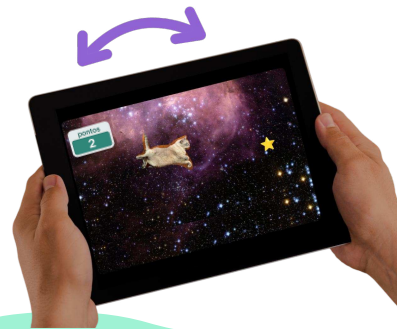
Clique para adicionar texto

Você pode movê-lo e mudar o tamanho

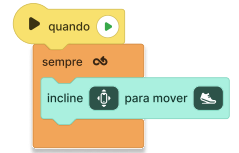
Fonte e cor

## Incline para mover

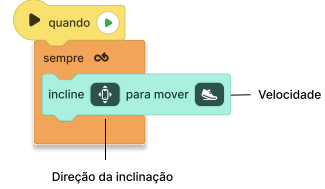
Controle um ator ao mover o aparelho



## Tente este código



## Explore mais

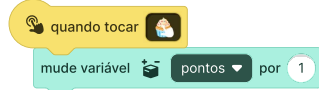


## Adicione pontuação

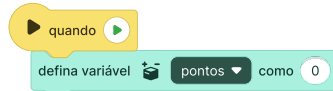
Adicione pontuação para criar um jogo



## Tente este código



## Para zerar a pontuação



## Emita um sinal

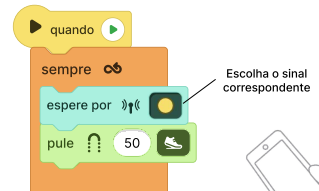
Use o bluetooth para interagir com amigos em outros tablets ou smartphones



## Emita um sinal



## Espera pelo sinal emitido em outro aparelho



Escolha o sinal correspondente

## Dicas

Pressione Play em todos os projetos para testá-los. Você pode adicionar um canal para transmitir, obtendo mais opções.