



## Programação visual no ensino fundamental: uso do Scratch

Ana Marli Bulegon  
Vanice Hentges

**Resumo:** Este trabalho apresenta os resultados parciais de um estudo de caso que envolve a programação visual no Ensino Fundamental a partir do uso do *Scratch*. Este estudo, desenvolvido no âmbito do curso de graduação em Sistemas de Informação, da Antonio Meneghetti Faculdade (AMF), teve como objetivo desenvolver conhecimentos básicos de Matemática de uma forma lúdica, aliando a programação de software às atividades de sala de aula. Os dados foram coletados a partir do uso do *Scratch* para o desenvolvimento de animações, jogos, histórias, etc. com crianças de nove a 11 anos de uma escola municipal em São João do Polêsine/RS. Os resultados obtidos demonstram que a programação de software potencializou o estudo de sala de aula e a aprendizagem de conceitos de matemática; despertou a autonomia das crianças para estudos extraclasse com e sobre a tecnologia de informação e comunicação; entre outros.

**Palavras-chave:** scratch; programação de software; cultura digital; ambiente de autoria.

### Visual programming in fundamental teaching: use of Scratch

**Abstract:** This paper presents the results of a case study involving the visual programming in Elementary Education from the Scratch use. This study, developed under the undergraduate degree in Information Systems, Antonio Meneghetti Faculdade (AMF), aimed to develop basic knowledge of mathematics in a playful way, combining software development to classroom activities. Data were collected from the use of Scratch for the development of animations, games, stories, etc. with children from 9 to 11 years old of a municipal school in São João do Polêsine/RS. The results show that software development has enhanced classroom study and learning of math concepts; He aroused the autonomy of children to extracurricular studies and the information and communication technology; among others.

**Keywords:** scratch; software development; digital culture; authoring environment.

### Programación visual en la enseñanza fundamental: uso de Scratch

**Resumen:** Este trabajo presenta los resultados parciales de un estudio de caso que involucra programación visual en la Enseñanza Fundamental a partir del uso de Scratch. Este estudio, desarrollado en el ámbito del curso de grado en Sistemas de Información, por la Facultad Antonio Meneghetti (AMF), tuvo como objetivo desarrollar conocimientos básicos de Matemáticas de una manera lúdica, combinando la programación de software con las actividades en el aula. Los datos fueron recolectados a partir del uso de Scratch para el desarrollo de animaciones, juegos, historias, etc. con niños de 9 a 11 años de edad de una escuela municipal en São João do Polêsine / RS. Los resultados obtenidos demuestran que la programación de software ha mejorado el estudio del aula y el aprendizaje de conceptos matemáticos; despertó la autonomía de los niños para los estudios extraescolares con y sobre la tecnología de la información y la comunicación; entre otros.

**Palabras clave:** Scratch; programación de software; cultura digital; ambiente de autoría.

## 1 Introdução

A educação é reflexo dos momentos vividos historicamente. Hoje, em pleno século XXI, a informação está acessível a todos e em todos os lugares, de modo que é impossível pensar na escola como a alguns anos atrás.

Estatísticas brasileiras mostram que a demanda por profissionais de computação cresce exponencialmente a cada ano e as universidades não estão formando na mesma proporção. Para resolver este problema, faz-se necessário aumentar o número de interessados pelos cursos da área de computação nas universidades, mas isso passa primeiro pela introdução de conceitos de ciência da computação na educação básica (NUNNES, 2011, s/p).

No contexto escolar, percebe-se que algumas crianças, nascidas imersas no mundo tecnológico, têm apresentado dificuldades nas atividades como desenhar, copiar um texto de um quadro, fazer cálculos no papel. Para essas crianças, atividades desse tipo parecem ser uma tarefa um tanto quando complicada, visto que o papel e o lápis não interagem com eles como os aparelhos eletrônicos. Ter escolas equipadas com aparelhos eletrônicos não basta para melhorar essa realidade. É necessário que os professores tenham conhecimento das ferramentas que podem auxiliar no processo de ensino e aprendizagem.

Assim, trazer conceitos computacionais para a Educação Básica é fundamental para manter e desenvolver o raciocínio computacional nas crianças e despertar o interesse delas pela área da tecnologia.

O Ensino Fundamental no Brasil, desde o ano de 2005, tem a duração de nove anos e é obrigatória. Seus estudantes são crianças de seis a 14 anos de idade. Nessa etapa da Educação Básica, deve desenvolver-se a capacidade de aprendizado do estudante. O estudo da Matemática nessa etapa da vida escolar é imprescindível, pois é nela que se formam os conceitos básicos sobre esse conhecimento. Entretanto, os estudantes demonstram muitas dificuldades de aprendizagem.

Para tentar minimizar essa situação, os professores podem lançar mão do uso das TIC (Tecnologia da Informação e Comunicação). A Base Nacional Comum Curricular (BNCC) orienta para o ensino por competências e elencou dez competências gerais, definidas a partir das Diretrizes Curriculares Nacionais (DCN) para a Educação Básica, e dos conhecimentos, habilidades e valores essenciais para a vida do século XXI (BRASIL, 2016, p. 2).

Uma das competências é a Cultura digital, entendida como uma cultura para: “Compreender, utilizar e criar tecnologias digitais de forma crítica, significativa e ética” (BRASIL, 2016, p. 31) a fim de: “Comunicar-se, acessar e produzir informações e conhecimentos, resolver problemas e exercer protagonismo e autoria” (BRASIL, 2016, p. 31). Porém, existe ainda muita divergência sobre a utilização dessas ferramentas nos primeiros anos escolares.

Essa competência vem ao encontro da geração atual, chamada de “nativos digitais<sup>1</sup>”, que é protagonista de seu próprio conhecimento e necessita de um professor que seja mediador do conhecimento; alguém que possa auxiliá-lo no processo de ensino e aprendizagem. Nesse modelo de ensino, a concepção é de que o professor ensina, mas também aprende e o conhecimento é construído por ambos: professores e estudantes, tornando-se um processo mais rico e significativo, em que a produção de conhecimento acontece e enriquece o processo.

Diante disso, este trabalho propõe-se apresentar as contribuições da programação com o uso de softwares no processo de aprendizagem lógico-matemático de estudantes do ensino fundamental a partir do software *Scratch* (<https://scratch.mit.edu/>).

## **2 Fundamentação Teórica**

### **2.1 Educação Básica no Brasil**

O desempenho do Brasil, em programas como o PISA (Programme for International Student Assessment – Programa Internacional de Avaliação de Estudantes), referentes aos conhecimentos de Matemática, Ciências e Leitura, na Educação Básica, está longe do ideal. As áreas mais deficitárias são as de Ciências e Matemática (TOKARNIA, 2014). De acordo com esses índices, em 2012, o Brasil apresentava um desempenho de 391 pontos, para Matemática; 410 para leitura e 405 para ciências (TOKARNIA, 2014). Em 2015, os índices eram: em ciências (401 pontos, comparados à média de 493 pontos), em leitura (407 pontos, comparados à média de 493 pontos) e em matemática (377 pontos, comparados à média de 490 pontos) (BRASIL, 2015).

O Pensamento Matemático é o tipo de pensamento que envolve o raciocínio lógico. É composto por conhecimentos de álgebra, geometria, lógica, entre outros, sendo muito utilizado nos processos de resolução de problemas, que, por meio de mecanismos de comparações, generalizações ou abstrações, o intelecto humano leva às respostas verdadeiras, falsas ou prováveis.

O Pensamento Computacional (PC), por sua vez, cujo termo foi criado em 2006 por Jeannete Wing (WING, 2006), expressa um tipo de pensamento que envolve habilidades de resolução de problemas, capacidade de projetar sistemas e a compreensão do comportamento humano, tem como base os conceitos da Ciência da Computação. Para Aho e Ullman (1992), o

---

<sup>1</sup> Nativos digitais – estudantes que nasceram imersos no mundo tecnológico. São aqueles “falantes nativos” da linguagem digital dos computadores, vídeo games e internet.

Pensamento Computacional “é uma ciência da abstração” e como tal, essa forma de pensamento consolida-se. A abstração aplicada pelo Pensamento Computacional dá o poder de dimensionar e lidar com a complexidade (WING, 2014). Para promover os conceitos relativos ao Pensamento Computacional, geralmente são utilizadas estratégias multidisciplinares, como a robótica, que, muitas vezes, é trabalhada na educação com os kits LEGO e/ou placa Arduino.

Isso traz um questionamento sobre metodologias de ensino e recursos didáticos utilizados no processo de ensino, na atualidade. Está na hora de rever esses conceitos e repensar as estratégias de ensino para despertar o interesse dos estudantes e potencializar a sua aprendizagem.

Aliado a isso, a BNCC detalha as dimensões e subdimensões das competências como pode-se ver a da Cultura Digital (Figura 01).

Figura 01 – Detalhamento da competência “Cultura Digital”

Competência	Dimensões	Subdimensões	
<p><b>5 Cultura Digital</b></p> <p><b>O que:</b> Compreender, utilizar e criar tecnologias digitais de forma crítica, significativa e ética</p> <p><b>Para:</b> Comunicar-se, acessar e produzir informações e conhecimentos, resolver problemas e exercer protagonismo e autoria</p>	COMPUTAÇÃO e PROGRAMAÇÃO	Utilização de ferramentas digitais	Utilização de ferramentas multimídia e periféricos para aprender e produzir.
		Produção multimídia	Utilização de recursos tecnológicos para desenhar, desenvolver, publicar, testar e apresentar produtos para demonstrar conhecimento e resolver problemas.
		Linguagens de programação	Utilização de linguagens de programação para solucionar problemas.
	PENSAMENTO COMPUTACIONAL	Domínio de algoritmos	Compreensão e escrita de algoritmos. Avaliação de vantagens e desvantagens de diferentes algoritmos. Utilização de classes, métodos, funções e parâmetros para dividir e resolver problemas.
		Visualização e análise de dados	Utilização de diferentes representações e abordagens para visualizar e analisar dados.
	CULTURA e MUNDO DIGITAL	Mundo digital	Compreensão do impacto das tecnologias na vida das pessoas e na sociedade, incluindo nas relações sociais, culturais e comerciais.
		Uso ético	Utilização das tecnologias, mídias e dispositivos de comunicação modernos de forma ética, comparando comportamentos adequados e inadequados.

Fonte: BNCC (BRASIL, 2016, p. 31).

Diante desse detalhamento, percebe-se que há muito a ser feito ainda para o desenvolvimento pleno da Cultura Digital, não só dos estudantes, mas também dos professores. O ensino ainda está baseado no uso de ferramentas digitais, seja para visualização de dados ou produção multimídia. Entretanto, ainda é preciso evoluir na utilização de linguagens de programação e no domínio de algoritmos para a solução de problemas básicos, oriundos do próprio uso da tecnologia, e para evolução e criação de novas tecnologias.

Nessa perspectiva, este trabalho justifica-se no sentido de melhorar a qualidade do ensino na área de Matemática, uma vez que se entende que o pensamento matemático é algo essencial para o desenvolvimento do Pensamento Computacional, base para a ampliação dessa competência e para a formação de cidadão do século XXI. Para atender os objetivos deste trabalho, esta pesquisa buscou mostrar que é possível ensinar programação para crianças, desenvolvendo o raciocínio lógico-matemático e também despertar o interesse pela área da tecnologia de forma lúdica e construtiva.

## 2.2 O conhecimento na atualidade e a informática na educação

Atualmente, vivencia-se um encontro de gerações com características muito diferentes, no sentido de conhecimento e de interação com a tecnologia; as crianças, nascidas

nos últimos 15 anos, já vêm imersas em um mundo tecnológico e que, com certeza, têm imensa facilidade em aprender e utilizar tudo que é tecnológico. Além disso:

O raciocínio computacional é intuitivo no ser humano e se manifesta já na idade infantil. Portanto, a criança naturalmente raciocina de forma computacional. Entretanto, tal fato não é explorado na formação básica. Como consequência, o raciocínio computacional intuitivo se perde ao longo do crescimento e da formação do indivíduo, a tal ponto que, em geral, um adolescente tem mais dificuldades de resolver problemas computacionais do que uma criança (NUNNES, 2011, s/p).

Esse fato é verificado porque a base do pensamento computacional é o raciocínio matemático, que, muitas vezes, é superficial, e do qual o Brasil carece de melhores resultados, como se pode verificar no desempenho das avaliações realizadas em instrumentos de avaliação internacionais, como nos índices do PISA, dos anos de 2012 e 2015. De acordo com esses índices, na área de Matemática, o Brasil está abaixo da média dos estudantes dos países que compõem a OCDE<sup>2</sup>, (377 pontos, comparados à média de 490 pontos). Verificou-se que, apesar do aumento de 21 pontos na média entre os anos de 2003 e 2015, houve um declínio de 11 pontos se for comparada a média de 2012 à média de 2015 (BRASIL/INEP, 2015, p.1).

Neste sentido, necessita-se de uma escola que promova uma aprendizagem significativa e eficaz na área de Matemática, que atenda essa demanda do conhecimento computacional aliada ao perfil da geração de estudantes do século XXI, que já está imersa na sociedade da tecnologia. A sugestão, neste caso, é que as atividades propostas gerem momentos de aprendizagem interativos. Não existe mais espaço para decorar conteúdos. Existe, sim, uma necessidade de produtores de conhecimento e, nos dias atuais, vivencia-se uma “chuva de informações”, em que saber o que acontece é algo simples, mas construir conhecimento, algumas vezes, parece distante da realidade. Assim, trabalhar com a programação visual, na Educação Básica, é fundamental para manter e desenvolver o raciocínio computacional.

Esse processo é bastante complexo e exige uma disponibilidade de professores, que não são “nativos digitais”, em entrar no mundo digital e perceber de que forma ele pode ser um facilitador nesse processo permitindo que o estudante utilize as tecnologias que lhe são tão comuns de forma produtiva, ou seja, produzindo conhecimento, conforme expressa Schlemmer (2006, p. 35).

Isso explica o motivo pelo qual muitos de nós ainda apresentam uma forma um tanto quanto enviesada de se relacionar com esses meios, o que é facilmente evidenciado quando e-mails e textos são impressos para serem lidos, ou, após serem

---

<sup>2</sup> OCDE - Organização para a Cooperação e Desenvolvimento Econômico

encaminhados, liga-se para saber se o sujeito recebeu. Isso faz com que pareçamos estrangeiros em nosso próprio mundo, como alguém que tenta falar a 'língua digital', mas com um forte sotaque analógico

Neste sentido, a utilização de recursos das TIC, no âmbito educacional, apresenta-se neste momento, como um grande desafio para muitos educadores. Selecionar esse recurso didático para o desenvolvimento de suas aulas exige do educador algumas atitudes, dentre elas, a mudança de postura didática, já que o tradicional e o novo não se complementam nesse cenário. Dessa forma:

[...] a escola precisa se deslocar das concepções de ensino/ aprendizagem, nas quais o livro e ela própria se configuram como únicas possibilidades de aquisição de conhecimento e de cultura [...] em direção a outras concepções, em que conhecimento, cultura e comunicação se aproximam, na medida em que são pensados a partir de novos parâmetros teórico/ conceituais (MARASCHIN e NEVADO, 1994, p. 782).

Diante disso, o ensino deve priorizar o estudo de conteúdos que possibilitem ao estudante interpretar criticamente o mundo que vive, permitindo que a Matemática torne-se mais agradável e desafiadora para uma compreensão mais fundamentada e consistente. Para isso, cabe à escola o desafio de transformar-se em espaço e agente de definição e articulação do que aprender e ensinar, na busca por incorporar ao ensino saberes que são oriundos do cotidiano.

Com esse pensamento, acredita-se que propondo atividades de aprendizagem interativas sobre programação visual os estudantes estarão construindo o conhecimento dos conceitos de Matemática de forma significativa, de modo a desenvolver o Pensamento Matemático por meio do Pensamento computacional.

### **2.3 Scratch**

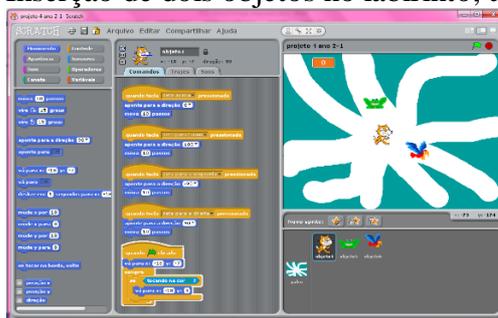
O *Scratch* é uma linguagem de programação visual, que foi desenvolvida pela equipe do Lifelong Kindergarten do MIT<sup>3</sup> Media Lab, (Massachusetts Institute of Technology), em 2007. A Figura 02 (dois) apresenta uma tela do *Scratch* para criação das animações. Na coluna da esquerda, são apresentados blocos de informações com linguagem de programação que, quando combinadas, geram ações dos objetos escolhidos como personagens da

---

<sup>3</sup> Massachusetts Institute of Technology – Estados Unidos da América.

animação. Na coluna central, aparece a combinação dos blocos com a linguagem de programação e, na coluna da direita, é possível visualizar os objetos escolhidos com a respectiva ação que foi programada na coluna central.

Figura 02: Inserção de dois objetos no labirinto, além do ator



Fonte: dados da pesquisa

A ideia do *Scratch* é que qualquer pessoa de qualquer idade possa programar, e ele deixa o utilizador com o comando das ações e interações. Programar usando o *Scratch* é mais fácil do que com linguagens tradicionais de programação, pois a base *Scratch* é juntar blocos, como se fosse um brinquedo do kit LEGO; basta arrastar os blocos com as ações pretendidas.

O *Scratch* foi desenvolvido com o intuito de auxiliar na construção de histórias interativas, músicas, jogos, animações e o que mais a criatividade permitir. Na base do *Scratch*, estão conceitos matemáticos como coordenadas cartesianas, variáveis, números e também conceitos computacionais como condições, interações, que são construídas ao longo do conhecimento da programação. Condições no *Scratch* é um conceito básico, pois a parte de código no *Scratch* é construída com o encaixe de blocos de comando e, assim, quando existe uma condição no bloco de comando, o código abaixo só irá funcionar se ela for verdadeira. Segundo Papert e Resnick (1995, p. 24):

Ao criarem seus projetos em Scratch, os jovens aprendem muitas habilidades do século XXI que serão críticas para um futuro de sucesso: pensar criativamente, comunicar-se claramente, analisar sistematicamente, usar tecnologias fluentemente, colaborar efetivamente, projetar iterativamente e aprender continuamente.

O *Scratch* surge com o intuito de valorizar o aprendizado pela experimentação, o aprender fazendo, o que pressupõe uma pedagogia que desenvolva a autonomia dos estudantes por meio do protagonismo e a autoria deles no processo de aprendizagem.

### 3 Metodologia

O trabalho aqui relatado é um recorte da pesquisa realizada para compor o Trabalho de Conclusão de Curso (TCC) da autora e sua orientadora, no curso de graduação em Sistemas de Informação, da Antonio Meneghetti Faculdade (AMF).

A pesquisa teve abordagem qualitativa, pois houve a preocupação com a compreensão do comportamento do grupo pesquisado, quando do uso da programação visual. Trata-se de um estudo de caso com estudantes do quarto e quinto anos do ensino fundamental de uma escola pública do interior do Rio Grande do Sul – Brasil, durante o ano de 2011. Na escola, as crianças das respectivas turmas tinham idades entre 10 e 13 anos.

As atividades do projeto “Descobrimo o Scratch” aconteciam durante o turno escolar, as aulas realizadas no laboratório de informática aconteciam duas vezes por semana com cada turma, com duração de 60 minutos para a turma do quinto ano, e a turma do quarto ano foi dividida para que as atividades fossem realizadas de forma individual em cada computador, em função do número de computadores do laboratório. O Quadro 01 (um) apresenta as atividades de aprendizagem desenvolvidas com o *Scratch*.

Quadro 01: Arquitetura pedagógica das aulas

<b>Projeto /aulas</b>	<b>Atividades desenvolvidas:</b>
Aula 1 – Apresentação do projeto e primeiras noções	<b>Atividades desenvolvidas:</b> - Questionário Inicial - Apresentação e explicação sobre o funcionamento do ambiente do <i>Scratch</i> (principais comandos e informações importantes para a utilização do ambiente); - Exploração do <i>Scratch</i> pelos estudantes.
Aula 2 – Apresentação e Experimentação do <i>Scratch</i> e desenvolvimento de animações e jogos pelos estudantes.	<b>Atividades desenvolvidas:</b> - Apresentação das ferramentas (controles e movimento) do <i>Scratch</i> ; - Demonstração de um exemplo com as ferramentas (controles e movimento); -Exploração do <i>Scratch</i> pelos estudantes.
Aula 3 – Apresentação e experimentação das ferramentas do <i>Scratch</i> que permitem dar movimento às animações.	<b>Atividades desenvolvidas:</b> - Apresentação das ferramentas (sons, aparência e sensores) do <i>Scratch</i> ; - Demonstração de um exemplo com as ferramentas (sons, aparência e sensores); -Exploração do <i>Scratch</i> pelos estudantes.
Aula 4 – Desenvolvimento de um jogo no <i>Scratch</i> pelos estudantes.	<b>Atividades desenvolvidas:</b> - Elaboração de uma aplicação com <i>Scratch</i> pelos estudantes.
Aula 5 - Desenvolvimento de um jogo no <i>Scratch</i> pelos estudantes.	<b>Atividades desenvolvidas:</b> -Elaboração de uma aplicação com <i>Scratch</i> pelos estudantes.

Aula 6 – Avaliação da aprendizagem.	<b>Atividades desenvolvidas:</b> -Construção de uma aplicação livre no <i>Scratch</i> . -Feedback das atividades. -Questionário final.
-------------------------------------	---

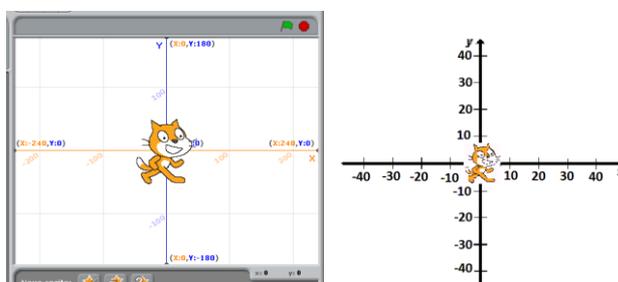
Fonte: as autoras.

A coleta de dados aconteceu em dois momentos: por meio de questionários e de observações do comportamento dos estudantes nas aulas com o uso do computador. Os questionários tinham como objetivo verificar o conhecimento inicial e final dos estudantes quanto ao uso do computador em sala de aula. As observações referiam-se à análise do comportamento dos estudantes diante do uso do computador em sala de aula e de seu conhecimento sobre como programar, utilizando uma linguagem de programação com blocos lógicos e de “*drag and drop*”<sup>4</sup> (arrastar e soltar). Usou-se o instrumento observação, pois este permite perceber como os estudantes interagem com o computador e desenvolvem a programação nele.

### 3.1 Conhecimentos de Matemática estudados

Os documentos oficiais sobre o currículo escolar indicam que, no quarto e quinto ano do Ensino Fundamental (etapa final dos Anos iniciais), o estudo de Matemática compreende os conceitos de Álgebra e Geometria de uma forma básica. No caso da Geometria, os estudantes devem ter domínio sobre os conceitos de Plano Cartesiano, que é um sistema de coordenadas cartesianas para demonstrar a localização de alguns pontos no espaço, criado por Descartes. Ele consiste no cruzamento de dois eixos, que pertencem a um plano em comum, de modo a serem perpendiculares (Figura 3).

Figura 03: Possibilidades do Scratch no ensino de Plano Cartesiano



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

<sup>4</sup> *Drag and drop*, tradução arrastar e soltar, linguagem utilizada quando se fala de linguagens gráficas de programação como é o caso do *Scratch*.

A linha vertical é chamada de eixo das ordenadas (y) e a linha horizontal, eixo das abscissas (x). Com a intersecção dessas linhas, tem-se a formação de quatro quadrantes, sendo que os números positivos vão para cima ou para a direita e os números negativos vão para a esquerda ou para baixo.

Ocorre que, na vida cotidiana, o conceito de negativo é de difícil compreensão. Entretanto, com a construção do movimento do personagem, no *Scratch* (Figura 3), em que as coordenadas cartesianas e as variáveis ficam visíveis aos usuários, os conceitos de negativo e positivo no Plano cartesiano tornam-se mais significativos e de fácil compreensão.

#### 4 Resultados e discussões

Os estudantes estiveram atentos a todas as informações dadas pela professora pesquisadora e interagiram com o *Scratch* de forma curiosa. Buscavam descobrir a infinidade de atores (personagens) que estão disponíveis no *Scratch*.

As figuras 04 (quatro) e 05 (cinco) apresentam os resultados da interação dos estudantes com o *Scratch*. Apesar da falta de habilidade dos estudantes com o *Scratch* e com os recursos do computador, pode-se perceber, por essas figuras, que eles desenvolveram habilidades, antes desconhecidas suas, como é a linguagem de programação implícita no software *Scratch*.

Figura 04: Simulação de um aquário, atividade realizada pelos estudantes



Fonte: dados da pesquisa.

A linguagem utilizada (coluna central do *Scratch*) - Figura 05 (cinco) permitiu que o peixe (objeto utilizado) tivesse movimentos e deslocamentos no *layout* (palco) escolhido.

Figura 05: Exemplo sobre como operar com o Scratch



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

Para compor esse movimento, o tipo de pensamento envolvido é o raciocínio lógico, aliado ao pensamento computacional.

Quanto aos conceitos de Matemática, a professora relata que os conceitos de Plano cartesiano, espaço físico, latitude, altitude, números positivos e negativos, além do raciocínio lógico, foram evidenciados pelos estudantes quando ocorreu a resolução de problemas de lápis e papel, em sala de aula. Esses conceitos estavam envolvidos na navegação no palco do *Scratch*. As Figuras 04 (quatro) e 05 (cinco) mostram que os estudantes necessitavam fazer escolhas de navegação e, com isso, os conceitos de coordenadas cartesianas foram trabalhados de forma lúdica e com significado para eles.

Além desses resultados, frequentemente, os estudantes relatavam o seu envolvimento com o *Scratch* em ambientes extraclasse. Isso denota a sua motivação e envolvimento com o *software* e as atividades propostas.

## 5 Considerações finais

Para que fosse possível entender de que forma a programação de *software* pode contribuir para a aprendizagem, incluiu-se a programação no ambiente escolar, por meio do *Scratch*. Esse *software* foi escolhido por ter uma interface interativa e apresentar uma linguagem visual, permitindo que crianças compreendessem e utilizassem essa ferramenta computacional sem grandes dificuldades. Com esse recurso, a programação acontece de forma lúdica, a partir da junção de blocos com códigos, o que facilita o entendimento das crianças e permite que desenvolvam os seus próprios projetos.

Ao utilizar o *Scratch*, os estudantes tornaram-se construtores de pequenas aplicações e trabalharam conceitos novos para eles, tanto no caso de Matemática como de Informática. Além disso, o *Scratch* despertou-lhes maior interesse sobre as tecnologias que estão presentes no dia-a-dia. Ao rever o que foi aprendido em aula, em espaços extraclasse, os estudantes praticavam em casa o que haviam aprendido na escola, repetindo, criando e recriando. Essas

informações permitem entender que a maioria dos estudantes tem facilidade em trabalhar com o computador e que se for explorado de forma lúdica e respeitado o protagonismo dos estudantes, pode-se colher ótimos resultados.

Por essas constatações, pode-se concluir que o ensino da programação de *softwares* na idade infantil contribui para o desenvolvimento do espírito investigativo e potencializa a aprendizagem no contexto escolar.

Ensinar programação para crianças de nove a 13 anos é, com certeza, um desafio, mas é imensamente gratificante ver os resultados; perceber que se pode trazer perspectivas diferentes para crianças de uma comunidade simples, mostrando as possibilidades de aprendizagem de Matemática.

Vê-se a programação ainda distante do cotidiano das escolas, porém cabe a educadores e profissionais da TI mostrar que é possível repensar a prática e tornar a escola mais atrativa e produtiva para os estudantes, fazendo uso de tecnologia como um apoio, agregando valor ao ensino e à pesquisa, mostrando que é necessário evoluir e construir o conhecimento.

## Referências

AHO, A. V., ULLMAN, J. D. **Foundations of Computer Science**. New York: W. H. Freeman and Company. 1992.

BRASIL. Ministério da Educação. Secretaria da Educação Básica. **Base Nacional comum curricular - Ensino Fundamental**. Brasília: DF, Dez, 2016. Disponível em: [http://movimentopelabase.org.br/wp-content/uploads/2018/03/BNCC\\_Competencias\\_Progressao.pdf](http://movimentopelabase.org.br/wp-content/uploads/2018/03/BNCC_Competencias_Progressao.pdf). Acesso em: 20 fev. 2019.

BRASIL/MEC/INEP – Resultados do PISA. Relatório OCDE. 2015 Disponível em [http://download.inep.gov.br/acoes\\_internacionais/pisa/resultados/2015/pisa\\_2015\\_brazil\\_pt.pdf](http://download.inep.gov.br/acoes_internacionais/pisa/resultados/2015/pisa_2015_brazil_pt.pdf). Acesso em: 25 jan. 2019.

MARASCHIN, C.; NEVADO, R. A. O Paradigma epistemológico e o ambiente de aprendizagem Logo. In: BRASIL. Ministério da Educação. Secretaria de Educação Média e Tecnológica. Informática na Escola: Pesquisas e Experiências. Léa da Cruz Fagundes (org.). Brasília: MEC/SEMTEC, 1994.

NUNNES, D. J., **Ciência da Computação na Educação Básica**. Blog ADUFRGS Sindical. Instituto de Informática da UFRGS. 2011. Disponível em: <http://www.adufrgs.org.br/artigos/ciencia-da-computacao-na-educacao-basica/>. Acesso em: 06 out. 2014.

TOKARNIA, M. **O Brasil investe metade do recomendado pela OCDE para o ensino básico**. 2014 Disponível em: <http://www.fundaplub.org.br/site/wp-content/uploads/2013/12/ranking-pisa-2012-2.jpg>. Acesso em: 26 jan. 2019.

PAPERT, S.; RESNICK, M. Technological Fluency and the Representation of Knowledge. Proposal to the National Science Foundation. MIT MediaLab. 1995.

SCHLEMMER, E., **Revista Textual**. Sindicato dos Professores do Estado do Rio Grande do Sul. Porto Alegre. set. 2006

WING, J. M. **Computational Thinking**. In Communications of the ACM, 2006. pp. 33–35.

WING, J. M. **Computational Thinking Benefits Society**. 2014. Disponível em: <http://socialissues.cs.toronto.edu/2014/01/computational-thinking/>. Acesso em: mar. 2019.