

ENTRE EIXOS E ALGORITMOS: MODELAGEM E SIMULAÇÃO DE UM BRAÇO ROBÓTICO COM SCRATCH, TINKERCAD E GEOGEBRA

BETWEEN AXES AND ALGORITHMS: MODELING AND SIMULATION OF A ROBOTIC ARM WITH SCRATCH, TINKERCAD, AND GEOGEBRA

ADENIR DOS SANTOS CAMARGO, FLÁVIA GONÇALVES
FERNANDES

RESUMO

Este artigo apresenta o desenvolvimento técnico de uma proposta para o ensino de Matemática na Educação Básica baseada na modelagem e simulação de um braço robótico com Scratch, Tinkercad e GeoGebra. A proposta integra programação por blocos, simulação eletrônica e modelagem geométrica dinâmica para representar um mesmo projeto sob diferentes perspectivas conceituais e operacionais. No Scratch, foram simulados movimentos articulados por meio de botões, variações e elementos gráficos, permitindo representar deslocamentos e rotações do braço robótico. No Tinkercad, foi construído um circuito simulado com cinco servomotores e cinco potenciômetros, possibilitando testar a lógica de controle dos movimentos e o mapeamento analógico-digital dos ângulos. No GeoGebra, a modelagem dinâmica foi deixada explícita relações geométricas envolvidas nos movimentos, com uso de vetores, translações, rotações e controles deslizantes. Os resultados indicam que a integração entre essas três plataformas é técnica viável e oferece potencial pedagógico para abordagens interdisciplinares envolvendo geometria, trigonometria, eletrônica básica e pensamento computacional. Como limitação, a proposta ainda não foi aplicada com estudantes, o que impede conclusões sobre aprendizagem ou impacto pedagógico em contexto real.

Palavras-chave: Tecnologias Digitais na Educação; Robótica Educacional; Metodologias Ativas.

ABSTRACT

This article presents the technical development of a proposal for teaching mathematics in K-12 education, based on the modeling and simulation of a robotic arm using Scratch, Tinkercad, and GeoGebra. The proposal integrates block-based programming, electronic simulation, and dynamic geometric modeling to represent a single project from different conceptual and operational perspectives. In Scratch, articulated movements were simulated using buttons, variables, and graphical elements, enabling the representation of the robotic arm's displacements and rotations. In Tinkercad, a simulated circuit featuring five servomotors and five potentiometers was constructed, allowing for the testing of movement control logic and the

analog-to-digital mapping of angles. In GeoGebra, dynamic modeling made the geometric relationships involved in the movements explicit through the use of vectors, translations, rotations, and sliders. The results indicate that integrating these three platforms is technically feasible and offers pedagogical potential for interdisciplinary approaches involving geometry, trigonometry, basic electronics, and computational thinking. A limitation of the study is that the proposal has not yet been implemented with students, precluding conclusions regarding learning outcomes or pedagogical impact in a real-world setting.

Keywords: Digital Technologies in Education; Educational Robotics; Active Methodologies.

INTRODUÇÃO

O ensino de Matemática na Educação Básica pode ser enriquecido por propostas que articulem conceitos matemáticos, tecnologias digitais e resolução de problemas em contextos significativos. Nesse cenário, a robótica educacional se destaca por favorecer a integração entre diferentes áreas do conhecimento e por oferecer situações de experimentação, modelagem e construção de artefatos, aproximando conteúdos abstratos de aplicações mais concretas (Abrantes; Serrazina; Oliveira; 1999).

A incorporação de recursos digitais ao ensino também responde à necessidade de ampliar as formas de representação e exploração de objetos matemáticos, especialmente em atividades que envolvem movimento, proporcionalidade, ângulos e transformações geométricas. Entre as possibilidades disponíveis, o Scratch favorece a programação por blocos e a simulação visual; o Tinkercad permite a experimentação eletrônica em ambiente virtual; e o GeoGebra contribui para a modelagem geométrica dinâmica de movimentos articulados (Ribeiro, 2006).

A integração dessas três plataformas é particularmente relevante porque permite representar um mesmo planejamento por perspectivas complementares: programação, simulação eletrônica e análise geométrica. No caso de um braço robótico, essa articulação cria condições para explorar, em uma mesma proposta, noções de coordenadas, rotações, vetores, controle de movimento e relações entre entradas e saídas.

Este trabalho apresenta o desenvolvimento técnico de uma proposta didática baseada na modelagem e simulação de um braço robótico com Scratch, Tinkercad e GeoGebra. O objetivo é descrever a construção do modelo nas três plataformas e

discutir suas potencialidades para o ensino interdisciplinar de Matemática, com ênfase na integração entre programação, simulação eletrônica e modelagem geométrica.

Trata-se, portanto, de uma proposta de desenvolvimento técnico ainda não aplicada com estudantes, o que impede inferências sobre aprendizagem ou impacto pedagógico em contexto real. Ainda assim, os modelos produzidos permitem uma avaliação inicial da técnica de integração entre as três plataformas e do seu potencial pedagógico como apoio ao ensino de Matemática na Educação Básica.

REFERENCIAL TEÓRICO

Robótica Educacional e Aprendizagem Ativa

A robótica educacional tem se consolidado como uma estratégia pedagógica capaz de promover a aprendizagem ativa por meio da resolução de problemas, da experimentação e da construção de artefatos tecnológicos. A proposta dialoga diretamente com a perspectiva construcionista de Seymour Papert (2008), segundo a qual os estudantes aprendem melhor quando constroem algo significativo, manipulando objetos concretos e digitais enquanto refletem sobre os processos envolvidos. Nesse sentido, a robótica oferece um ambiente rico para a exploração de conceitos matemáticos, físicos e computacionais de forma integrada.

Ribeiro (2006) destaca que o entusiasmo das crianças ao trabalhar com robôs pode ser canalizado para fins educativos, ampliando o engajamento e o protagonismo discente. A robótica, portanto, atua como mediadora entre teoria e prática, permitindo que o estudante vivencie situações autênticas de investigação, planejamento e tomada de decisão. Campos (2019) e César (2005) reforçam que ambientes baseados em robótica favorecem o desenvolvimento de habilidades socioemocionais, como colaboração, criatividade e autonomia, ao mesmo tempo em que aproximam os alunos da cultura digital e das práticas tecnológicas contemporâneas.

Além disso, a robótica contribui para o trabalho interdisciplinar ao integrar conhecimentos de matemática, eletrônica, física, algoritmos e design, conforme argumentam Borba e Penteado (2012). Essa característica torna a área especialmente relevante no contexto da Educação Básica, onde a superação de compartimentalizações curriculares é essencial para a compreensão global dos fenômenos. Câmara, Machado e Willians (2018) ressaltam ainda que a robótica

fomenta competências do século XXI, tais como a resolução criativa de problemas e o pensamento crítico, alinhando-se às diretrizes da Base Nacional Comum Curricular (BRASIL, 2018).

Dessa forma, a robótica educacional se apresenta como uma ferramenta potente para a aprendizagem ativa, ampliando as oportunidades de experimentação, investigação e construção de conhecimentos de maneira colaborativa e significativa.

Pensamento Computacional e Metodologias Ativas (Design Thinking)

O desenvolvimento do pensamento computacional tem recebido crescente atenção nas últimas décadas, sobretudo após a difusão do conceito proposta por Wing (2006), que o define como uma habilidade essencial para compreender, modelar e solucionar problemas de forma sistemática. Segundo a autora, pensar computacionalmente vai além da programação: envolve analisar, decompor, abstrair e criar algoritmos para responder a desafios complexos.

Estudos como os de Brackmann (2017) e Blikstein (2008) reforçam que atividades envolvendo computação e robótica favorecem o desenvolvimento do pensamento computacional na Educação Básica ao proporcionar situações práticas que exigem raciocínio lógico, tomada de decisão e refinamento de soluções. Quando integradas a metodologias ativas, essas práticas tornam-se ainda mais potentes, pois posicionam o estudante como protagonista do processo de criação.

Nesse contexto, o Design Thinking (DT) surge como uma abordagem metodológica relevante para organizar processos criativos baseados na empatia, na experimentação e na prototipagem (Brown, 2008; 2010). O DT valoriza a escuta ativa, a compreensão profunda do problema e a produção de soluções colaborativas e iterativas, aproximando-se das demandas educacionais contemporâneas. Plattner et al. (2011) afirmam que o DT promove ambientes de aprendizagem nos quais o erro é compreendido como parte do processo criativo, estimulando a persistência e a reflexão.

Ao integrar Design Thinking, programação e robótica, criam-se oportunidades para que os estudantes desenvolvam competências cognitivas e socioemocionais simultaneamente. Projetos que envolvem prototipagem digital e física, como o desenvolvimento de braços robóticos, mobilizam o pensamento computacional

enquanto exigem planejamento, criatividade e análise crítica do problema inicial. Assim, DT e computação reforçam-se mutuamente, configurando um ambiente fértil para o aprendizado significativo.

Tecnologias Digitais na Educação Matemática (Scratch, Tinkercad, GeoGebra)

O uso de tecnologias digitais na Educação Matemática tem se destacado como uma estratégia eficaz para promover visualização, experimentação e compreensão conceitual. Borba e Penteado (2012) argumentam que as tecnologias não são meros instrumentos auxiliares, mas agentes que transformam o próprio modo como o conhecimento matemático é produzido e compreendido.

Entre as plataformas mais utilizadas, o Scratch ocupa papel central na introdução da programação para crianças e jovens. Criado com base nos princípios do construcionismo, o Scratch permite a criação de algoritmos por meio de blocos visuais, favorecendo o desenvolvimento do raciocínio lógico e da compreensão de estruturas algorítmicas (Pereira, 2019). Além disso, a plataforma possibilita simulações de movimentos, representações geométricas e interações dinâmicas, contribuindo para a aprendizagem de conceitos matemáticos, como coordenadas, ângulos e proporção.

O Tinkercad, por sua vez, é uma ferramenta que possibilita simulação eletrônica acessível, permitindo que estudantes experimentem circuitos e controlem componentes como servomotores e sensores. Autores como De Medeiros e Wünsch (2019) destacam que a plataforma aproxima os alunos do universo da eletrônica e da robótica de forma segura e prática, possibilitando testes sem necessidade de laboratórios físicos. A modelagem de circuitos apoiada em noções matemáticas, como ângulos de rotação, proporcionalidade e grandezas elétricas, fortalece a interdisciplinaridade entre Matemática, Física e Tecnologia.

O GeoGebra, amplamente reconhecido como ferramenta dinâmica para o ensino de Geometria Plana e Analítica (MARCHETTI; KLAUS, 2014; Pacheco, 2019), oferece recursos para visualização e manipulação de objetos geométricos em tempo real. Sua capacidade de trabalhar com controles deslizantes, vetores, translações e rotações torna-o especialmente adequado para representar movimentos articulados, como os de um braço robótico. Estudos como os de Silva (2017) e Fernandes (2013)

mostram que o GeoGebra favorece a compreensão de conceitos abstratos ao permitir que os estudantes observem relações matemáticas de forma dinâmica e exploratória.

A integração entre Scratch, Tinkercad e GeoGebra amplia significativamente o potencial pedagógico, pois cada plataforma contribui com perspectivas complementares:

- Scratch: programação e simulação visual;
- Tinkercad: experimentação eletrônica e controle físico;
- GeoGebra: modelagem geométrica e análise matemática.

Combinadas, essas tecnologias possibilitam a construção de sequências didáticas inovadoras, que fortalecem o pensamento computacional, o raciocínio matemático e a capacidade de aplicar conceitos de forma prática e contextualizada.

MATERIAIS E MÉTODOS

Trata-se de um estudo de desenvolvimento técnico, caracterizado como prova de conceito, voltado para modelagem e simulação de um braço robótico em três plataformas digitais: Scratch, Tinkercad e GeoGebra.

Foram utilizados os softwares Scratch, Tinkercad e GeoGebra para representar diferentes dimensões do modelo. No Tinkercad, foi usado um Arduino Uno R3, cinco servomotores e cinco potenciômetros, organizados para permitir o controle individualizado dos movimentos do braço robótico.

O desenvolvimento ocorreu em três etapas. Na primeira, foi construída uma simulação dos movimentos no Scratch. Na segunda, foi feito no Tinkercad o circuito virtual com os servomotores e potenciômetros. Na terceira, fez-se uma modelagem geométrica dinâmica no GeoGebra, com foco na representação dos movimentos articulados.

No circuito simulado no Tinkercad, o primeiro servo foi conectado à entrada A1 e à saída digital 2. Procedeu-se de forma analógica para os servos 2 a 5, conectados respectivamente às entradas A2–A5 e às saídas digitais 3–6. Para fins de apresentação, uma única figura representa esse padrão de ligação, evitando repetições desnecessárias.

A análise concentrou-se nas possibilidades técnicas de cada ambiente para representar movimentos, controle e relações matemáticas associadas ao braço robótico.

Programação no Scratch

Para a construção do Braço Robótico, primeiramente foi pesquisada a existência de outros materiais sobre braços Robóticos disponíveis no Scratch, o qual se destacou: <https://scratch.mit.edu/projects/547112196/>.

O modelo desenvolvido pelo usuário apresenta dois graus de liberdade, correspondentes aos movimentos do braço e do antebraço, além da manipulação da garra, totalizando três eixos de movimento. Na presente proposta, pretendemos desenvolver um braço robótico com cinco eixos, incluindo rotação horizontal em torno do eixo e rotação da garra. No entanto, como a representação desse movimento exigia recursos tridimensionais, o Scratch possui limitações nesse aspecto, as rotações em torno do eixo e não foi renovada. Em contrapartida, foi desligado o movimento de giro da garra, ampliando as possibilidades de rotação do modelo.

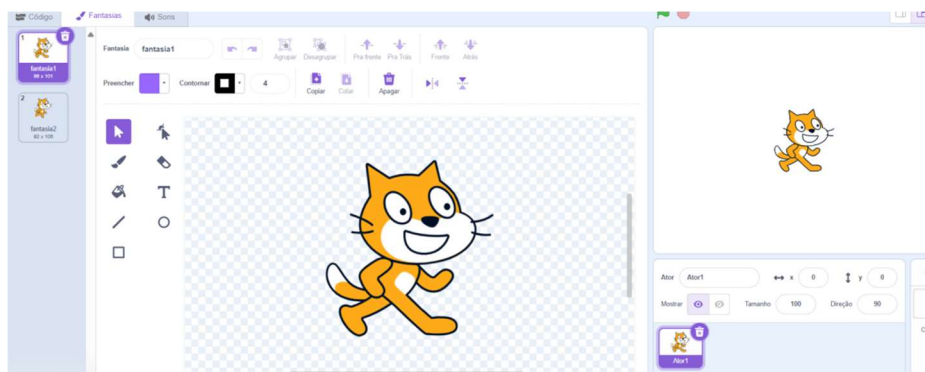
No projeto original, a movimentação dos eixos é realizada por meio do teclado. Nesta proposta, optou-se pelo uso de botões específicos para cada ação: esquerda, direita, cima, baixo, giro da garra para a direita, giro da garra para a esquerda, abrir a garra e fechar a garra. Além disso, o movimento foi programado para ocorrer apenas enquanto o botão correspondente estiver restrito, de modo a aproximar a simulação do funcionamento real de um braço robótico. Por esse motivo, o código preciso foi adaptado e, em vários casos, reconstruído, assim como alguns componentes gráficos, com o objetivo de tornar a proposta mais adequada ao contexto escolar.

O projeto original também inclui obstáculos e um objeto para manipulação. Nesta proposta, esses elementos não foram considerados, pois o foco é exclusivamente na construção e movimentação do braço robótico, podendo futuras versões incorporar novas funcionalidades.

Ao abrir a interface inicial do Scratch, observe como padrão apenas o “Scratch Cat”, mascote criado por Wing Ngan. Esse ator pode ser modificado, assim como suas fantasias, e também é possível inserir novos atores. A interface oferece recursos

intuitivos de edição e carregamento de elementos, o que facilita seu uso por iniciantes. A Figura 1 apresenta a mascote do Scratch e suas fantasias iniciais.

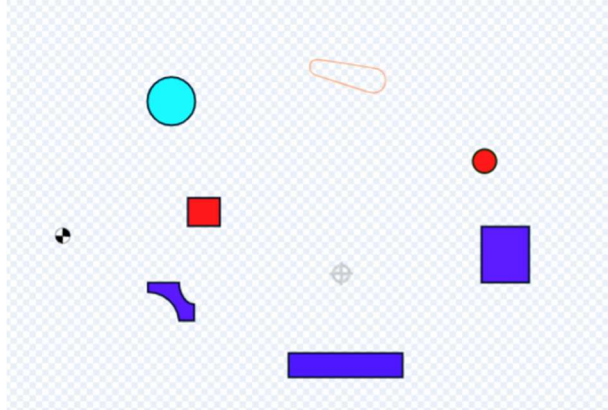
Figura 1 – Scratch Cat e Fantasias



Fonte: O Autor, (2024).

Para o desenvolvimento da proposta, foram criados os atores e suas respectivas fantasias, sendo, paralelamente, implementados os códigos responsáveis por sua movimentação. A interação com o braço robótico foi concebida para ocorrer por meio de botões de controle, iniciando-se pela implementação da base do mecanismo.

O Scratch disponibiliza uma biblioteca de atores predefinidos, além de um editor gráfico que permite a criação de novos elementos personalizados. Na construção da base do braço robótico, foram utilizadas as ferramentas de desenho do próprio ambiente, empregando principalmente formas geométricas, como círculos e retângulos, bem como recursos de preenchimento, contorno e remodelagem para a obtenção de bordas arredondadas e detalhes estruturais. A base foi projetada para representar um conjunto mecânico composto por motor, correia de transmissão e engrenagem móvel, simulando o sistema responsável pela rotação do braço robótico. O círculo em destaque representa o eixo de rotação do mecanismo. A Figura 2 apresenta os componentes utilizados na construção da base do braço robótico.

Figura 2 – Componentes da Base do Braço Robótico

Fonte: O Autor, (2024).

Após a integração das diferentes formas geométricas, obteve-se a estrutura completa da base do braço robótico, composta pela base, pelo motor, pelo suporte da engrenagem, pela correia de transmissão e pela engrenagem responsável pelo acionamento do mecanismo. Esses elementos foram organizados de modo a representar o sistema de transmissão do movimento e o eixo de rotação do braço robótico. A Figura 3 apresenta a composição final da base, evidenciando seus principais componentes estruturais e mecânicos.

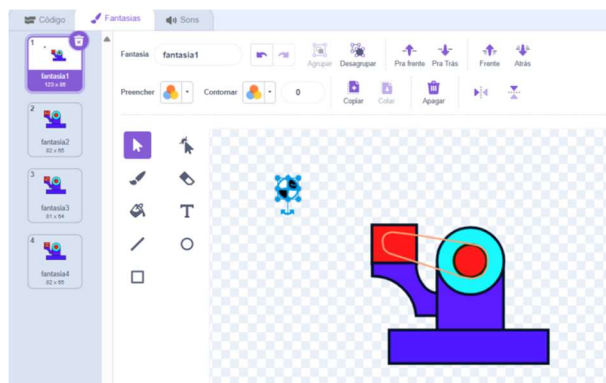
Figura 3 – Base do Braço Robótico Completa

Fonte: O Autor, (2024).

Uma estratégia para representar o movimento da base consiste na criação de diferentes fantasias da engrenagem, cada uma correspondente a uma posição angular distinta. Durante a execução da simulação, a alternância dessas fantasias em intervalos predefinidos produz a impressão visual de rotação, simulando o funcionamento do mecanismo de acionamento da base do braço robótico. A Figura 4

apresenta a construção das fantasias e as diferentes posições da engrenagem utilizadas para representar esse movimento.

Figura 4 – Construção das Fantasias no Braço Robótico



Fonte: O Autor, (2024).

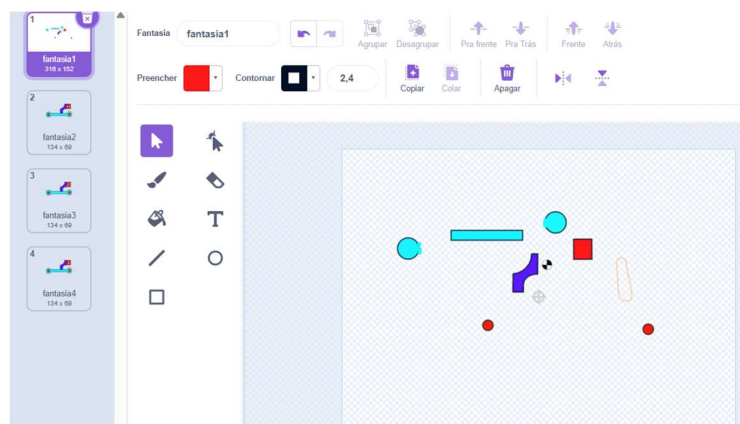
A janela do Scratch corresponde a um plano cartesiano no qual os objetos são posicionados por meio de coordenadas nos eixos horizontal e vertical. Esse sistema de referência deve ser considerado durante o desenvolvimento do braço robótico virtual, pois permite determinar com precisão a posição da base e dos demais componentes ao longo da simulação.

Para facilitar o processo de modelagem, foi estabelecida uma relação de proporcionalidade entre as dimensões físicas do protótipo e o sistema de coordenadas utilizado pelo Scratch. Essa conversão possibilitou representar, de forma consistente, as medidas reais dos componentes no ambiente virtual, preservando suas proporções e favorecendo a fidelidade da simulação.

A base do braço robótico foi inicialmente posicionada em um ponto de referência do plano cartesiano. Em seguida, foram implementadas funcionalidades que permitem seu deslocamento tanto no sentido horizontal quanto no vertical, simulando movimentos frequentemente observados em braços robóticos utilizados em aplicações reais. Para representar visualmente esse deslocamento, foram adicionados elementos gráficos, como rodas e mecanismos de sustentação. Também foram desenvolvidas rotinas para ocultar parcialmente a base quando está se aproxima dos limites da área de trabalho, permitindo futuras implementações relacionadas à detecção de colisões com obstáculos, paredes ou com o próprio solo.

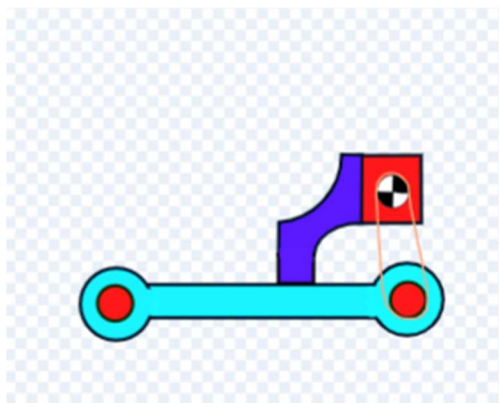
De maneira semelhante, foi construída a primeira seção do braço robótico, responsável pela ligação entre a base e a estrutura superior. Essa parte possui um ponto de articulação com a base e outro destinado à conexão com a segunda seção do mecanismo, representando as juntas responsáveis pelo movimento. Para proporcionar continuidade visual entre os componentes, foram utilizados recursos gráficos do Scratch, como o pincel e o agrupamento de diferentes formas geométricas, possibilitando a criação de fantasias que representam as diferentes posições de rotação da estrutura. A Figura 5 apresenta a construção dos componentes que compõem essa primeira seção do braço robótico.

Figura 5 – Componentes do Braço



Fonte: O Autor, (2024).

Após a integração de todos os componentes, obteve-se a primeira seção do braço robótico. Sua posição inicial foi definida de modo que o primeiro eixo de rotação coincidisse com o eixo de rotação da base, garantindo a correta articulação entre os dois elementos. Como essa estrutura também possui um segundo eixo de rotação, destinado à conexão com o antebraço, foi necessário realizar cálculos para determinar sua posição com precisão, considerando as dimensões da peça e a geometria do mecanismo. Esse procedimento assegura o alinhamento adequado entre as articulações e a fidelidade dos movimentos simulados. A Figura 6 apresenta a primeira seção do braço robótico completa, destacando ambos os eixos de rotação.

Figura 6 – Braço Completo

Fonte: O Autor, (2024).

O primeiro eixo de rotação do braço foi definido de forma coincidente com o eixo de rotação da base, considerando sua posição inicial no plano cartesiano e o eventual deslocamento horizontal da estrutura. A posição do segundo eixo de rotação, responsável pela articulação com o antebraço, foi determinada a partir das dimensões do braço, convertidas para o sistema de coordenadas do Scratch por meio da relação de proporcionalidade estabelecida entre as medidas físicas e os pixels do ambiente de programação.

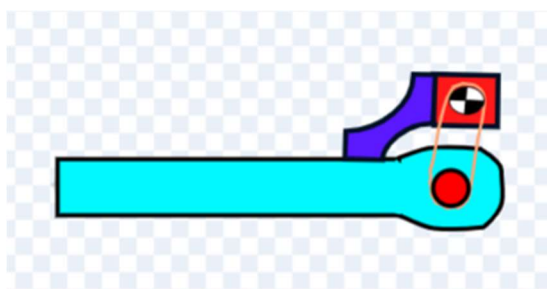
Como o braço robótico realiza movimentos de rotação, a posição do segundo eixo é atualizada continuamente em função do ângulo da primeira articulação. Para isso, empregou-se a decomposição vetorial com o uso das funções trigonométricas seno e cosseno, permitindo calcular dinamicamente as coordenadas do eixo de articulação do antebraço em relação à posição da base durante toda a movimentação do mecanismo.

Caso sejam implementados deslocamentos verticais da base, uma variável auxiliar poderá ser incorporada ao cálculo da coordenada vertical, possibilitando atualizar corretamente a posição de todas as articulações ao longo da simulação.

A construção do antebraço seguiu os mesmos princípios adotados para a base e para a primeira seção do braço robótico. Foram utilizadas as ferramentas de desenho do Scratch para modelar seus componentes estruturais, incluindo a engrenagem e a correia de transmissão, que posteriormente foram integrados em um único ator. Em sua extremidade foi posicionada a garra, cuja localização foi definida a partir do segundo eixo de rotação, garantindo o alinhamento entre as articulações e o

comportamento cinemático adequado durante a simulação. A Figura 7 apresenta o antebraço completo, destacando o eixo de rotação da garra e sua respectiva programação de movimentação.

Figura 7 – Antebraço Completo



Fonte: O Autor, (2024).

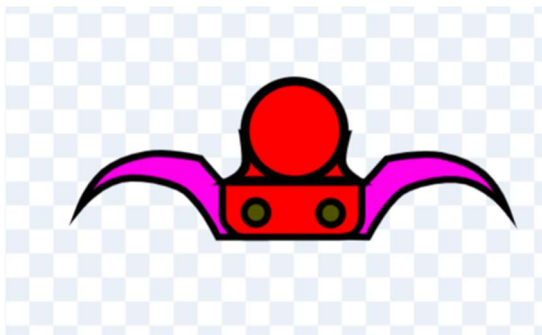
A posição inicial da garra foi determinada a partir da orientação do antebraço, considerando o segundo ângulo de rotação do mecanismo e as dimensões dessa estrutura. Para isso, foi utilizada a mesma relação de proporcionalidade empregada nas etapas anteriores, permitindo converter as medidas físicas para o sistema de coordenadas do Scratch.

Como a garra está acoplada à extremidade do antebraço, sua posição é atualizada continuamente durante a movimentação do braço robótico. Esse cálculo considera as coordenadas do eixo de rotação do antebraço e emprega funções trigonométricas para determinar a nova posição da garra em cada instante da simulação. Para facilitar a implementação, foram utilizadas variáveis auxiliares responsáveis por armazenar as coordenadas da articulação anterior, garantindo o correto posicionamento da garra ao longo dos movimentos.

A construção da garra seguiu os mesmos princípios adotados para os demais componentes do braço robótico. Utilizaram-se formas geométricas básicas disponíveis no Scratch, como retângulos e círculos, para representar sua estrutura. Como ponto de partida, foi utilizado o ator da construção original, realizando-se adaptações por meio da remoção do sensor, uma vez que esse elemento não fazia parte da proposta desenvolvida. Entretanto, a estrutura foi concebida de forma a permitir a futura implementação desse recurso, caso seja necessário ampliar as

funcionalidades da simulação. A Figura 8 apresenta a garra do braço robótico com sua modelagem e programação concluídas.

Figura 8 – Garra do Braço Robótico



Fonte: O Autor, (2024).

Após a construção dos componentes do braço robótico, foram desenvolvidos os botões responsáveis pelo controle de suas principais funcionalidades. A interface permite a movimentação horizontal da base, o acionamento das articulações do braço e do antebraço em ambos os sentidos, além da abertura, do fechamento e da rotação da garra. Esses comandos possibilitam a interação com o modelo virtual de forma intuitiva, simulando os movimentos básicos de um braço robótico.

Embora a proposta tenha sido desenvolvida em um ambiente bidimensional, o que limita a representação de alguns movimentos presentes em sistemas robóticos reais, a estrutura foi projetada de forma modular, permitindo a incorporação de novas funcionalidades em trabalhos futuros. Nesta versão, foram implementados os movimentos correspondentes aos motores responsáveis pela rotação da base, do braço, do antebraço e da garra, contemplando os principais graus de liberdade previstos para a simulação.

Como complemento ao modelo, foram adicionadas rodas à base do braço robótico, representando um sistema móvel capaz de realizar deslocamentos horizontais. A inclusão desse mecanismo exigiu adaptações na programação dos demais componentes, garantindo a sincronização dos movimentos e preservando o alinhamento entre as articulações durante toda a simulação.

Concluída a modelagem e a implementação da programação de todos os componentes, obteve-se um simulador funcional de braço robótico desenvolvido no

Scratch. Embora o modelo ainda possa receber aprimoramentos, como a incorporação de sensores, detecção de colisões e novos graus de liberdade, a versão atual atende aos objetivos propostos e constitui uma ferramenta para o ensino de conceitos relacionados à robótica, à programação e à matemática. A versão completa da simulação pode ser acessada no ambiente Scratch: <https://scratch.mit.edu/projects/1059396001/fullscreen/>

Simulação no Tinkercad

Para a construção do braço robótico no Tinkercad foi adotada uma arquitetura com cinco eixos articulados, permitindo a simulação dos principais movimentos do mecanismo. O circuito foi composto por um microcontrolador Arduino Uno, uma placa de prototipagem, potenciômetros destinados ao controle individual de cada articulação, micro servomotores responsáveis pelos movimentos dos eixos e cabos de conexão para a integração dos componentes.

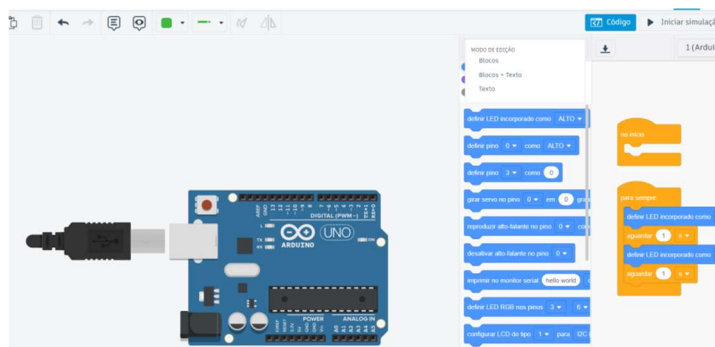
No ambiente Tinkercad, os potenciômetros possuem um valor de resistência padrão que pode ser alterado conforme as necessidades do projeto. Essa modificação influencia o comportamento elétrico do circuito, sendo sua análise fundamentada nos princípios básicos da eletricidade, especialmente na relação entre tensão, corrente elétrica e resistência.

O ambiente de desenvolvimento do Tinkercad disponibiliza uma biblioteca de componentes organizada por categorias, permitindo selecionar tanto os elementos mais utilizados quanto o conjunto completo de dispositivos eletrônicos disponíveis para simulação. Para este projeto, foi utilizado o Arduino Uno R3 como unidade de controle do sistema.

Após a inserção dos componentes eletrônicos no circuito, o ambiente de programação do Tinkercad permite a implementação do código de controle por meio da opção de edição em linguagem textual. Essa funcionalidade possibilita desenvolver algoritmos para o acionamento dos servomotores e a leitura dos potenciômetros, reproduzindo o comportamento esperado do braço robótico. Embora o ambiente também ofereça alternativas de programação em blocos, a programação textual foi adotada neste trabalho por proporcionar maior flexibilidade na implementação das rotinas de controle.

A Figura 9 apresenta a interface inicial do Tinkercad Circuitos, destacando a seleção do Arduino Uno R3 e o acesso ao ambiente de programação em linguagem textual.

Figura 9 – Interface Inicial e Código em Texto no Tinkercad Circuitos



Fonte: O Autor, (2024).

Ao acessar o ambiente de programação em linguagem textual, o Tinkercad apresenta automaticamente um código padrão contendo a estrutura básica de um programa para Arduino. Essa estrutura serve como ponto de partida para o desenvolvimento das rotinas de controle do circuito eletrônico.

A função `setup()` é responsável pela inicialização do sistema, configurando os componentes e definindo os pinos de entrada e saída do microcontrolador. Essa função é executada apenas uma vez, no início da execução do programa, sendo utilizada para preparar o circuito para seu funcionamento.

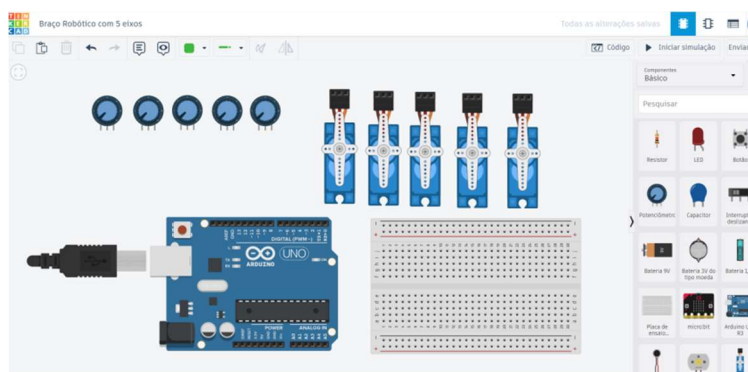
A função `loop()` contém as instruções executadas continuamente durante a simulação. Nela são implementadas as rotinas responsáveis pela leitura dos potenciômetros, pelo processamento das informações recebidas e pelo acionamento dos servomotores, permitindo a movimentação das articulações do braço robótico de forma dinâmica.

Durante a programação do Arduino, também são utilizadas funções para configurar o modo de operação dos pinos, controlar dispositivos conectados ao microcontrolador e estabelecer intervalos de tempo entre as instruções. Esses recursos são fundamentais para o correto funcionamento do sistema e para a sincronização dos movimentos do braço robótico.

Embora o Tinkercad forneça a estrutura inicial do programa, a implementação das rotinas de controle deve ser realizada pelo usuário de acordo com os objetivos do projeto. Dessa forma, enquanto não forem estabelecidas as conexões entre os componentes e desenvolvido o código correspondente, a execução da simulação não produzirá qualquer resposta do circuito.

A Figura 10 apresenta os principais componentes eletrônicos utilizados na construção do braço robótico no ambiente Tinkercad Circuitos. As conexões entre esses componentes são realizadas por meio de cabos de ligação, que são inseridos durante a montagem e a implementação da programação do sistema.

Figura 10 – Componentes Eletrônicos do Braço Robótico



Fonte: O Autor, (2024).

Inicialmente, foi incluída a biblioteca Servo, responsável por disponibilizar as funções necessárias ao controle dos servomotores utilizados no projeto. Em seguida, foi criado um objeto associado a essa biblioteca para realizar o acionamento do primeiro servomotor.

Posteriormente, foi declarada uma variável destinada ao armazenamento das leituras provenientes do potenciômetro. Na função de inicialização do programa, foram configurados os pinos de entrada e saída do Arduino utilizados na comunicação com o potenciômetro e com o servomotor, sendo este associado ao respectivo pino de controle.

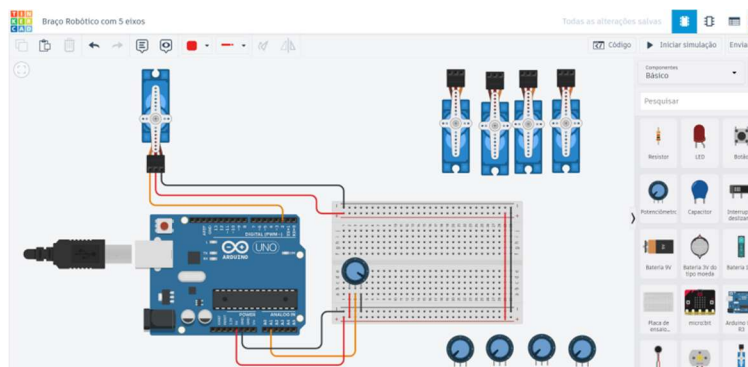
Na rotina principal do programa, a leitura do potenciômetro é realizada continuamente por meio da entrada analógica do microcontrolador. Os valores obtidos são convertidos para a faixa angular de operação do servomotor, permitindo que a

posição do eixo seja ajustada proporcionalmente ao deslocamento do potenciômetro. O ângulo calculado é armazenado na variável de controle e enviado ao servomotor, promovendo sua movimentação em tempo real.

Para a implementação do primeiro eixo articulado, foram definidos identificadores específicos para o servomotor e para a variável responsável pelo armazenamento da leitura do potenciômetro, bem como os pinos de entrada e saída correspondentes no Arduino. Esse procedimento foi posteriormente reproduzido para os demais eixos do braço robótico.

A Figura 11 apresenta a montagem do primeiro circuito de controle, evidenciando a conexão do sinal de controle do servomotor ao pino digital do Arduino e a ligação do potenciômetro à entrada analógica do microcontrolador. Também são apresentadas as conexões de alimentação e aterramento dos componentes, realizadas por meio da placa de prototipagem, assegurando o funcionamento adequado do circuito eletrônico.

Figura 11 – Associação de Um Servo Motor e Um Potenciômetro ao Braço Robótico



Fonte: O Autor, (2024).

A implementação dos demais eixos articulados do braço robótico seguiu a mesma metodologia adotada para o primeiro eixo. Para cada articulação, foi criado um novo objeto da biblioteca Servo, bem como uma variável destinada ao armazenamento das leituras do potenciômetro correspondente. Da mesma forma, foram configurados os respectivos pinos de entrada e saída do Arduino e realizadas as conexões individuais entre os componentes por meio da placa de prototipagem.

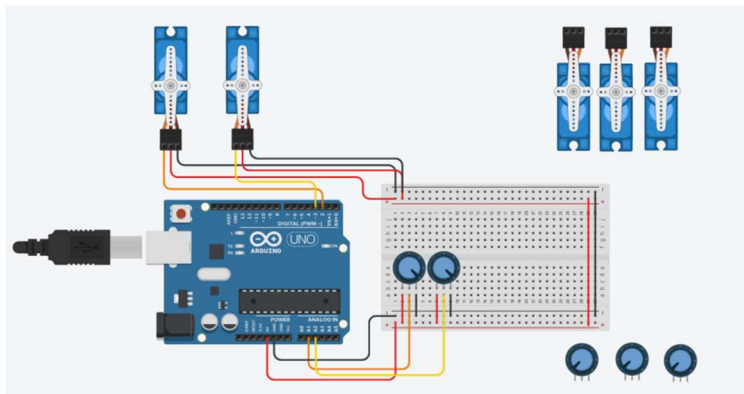
Na rotina principal do programa, as leituras provenientes de cada potenciômetro são continuamente convertidas para a faixa angular de operação do

respectivo servomotor. Os valores obtidos são armazenados nas variáveis de controle e utilizados para posicionar cada servo de forma proporcional ao movimento realizado pelo potenciômetro, permitindo o acionamento independente de cada articulação do braço robótico.

Para o segundo eixo, foram definidos um novo objeto para o servomotor, uma variável específica para armazenar os valores do potenciômetro e novos pinos de entrada e saída no Arduino, mantendo a mesma lógica de programação empregada na primeira articulação. Esse procedimento foi repetido para todos os eixos do mecanismo, alterando-se apenas os identificadores das variáveis, os objetos correspondentes aos servomotores e os pinos utilizados em cada conexão.

A Figura 12 apresenta a montagem do circuito correspondente ao segundo eixo articulado, evidenciando as conexões entre o servomotor, o potenciômetro, o Arduino e a placa de prototipagem, incluindo os terminais de alimentação, aterramento e sinal de controle necessários ao funcionamento do sistema.

Figura 12 – Associação de Dois Servos Motores e Dois Potenciômetros ao Braço Robótico



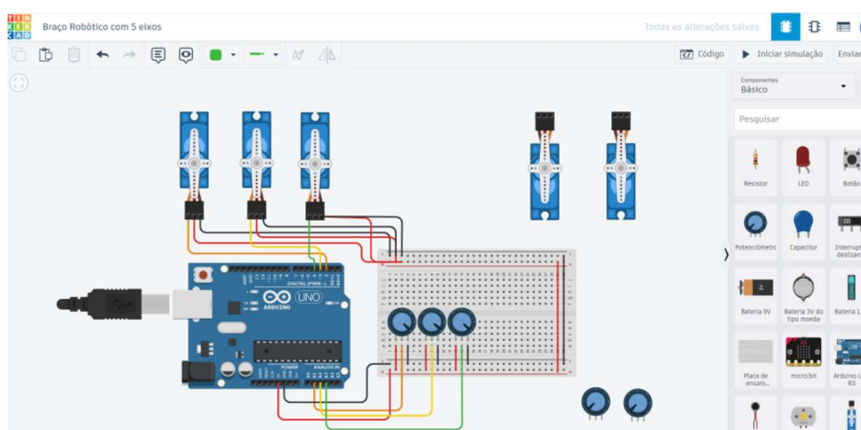
Fonte: O Autor, (2024).

A implementação do terceiro eixo articulado seguiu a mesma lógica empregada nas etapas anteriores. Para essa articulação, foi criado um novo objeto da biblioteca Servo, uma variável destinada ao armazenamento das leituras do terceiro potenciômetro e a configuração dos respectivos pinos de entrada e saída do Arduino. Na rotina principal do programa, a leitura do potenciômetro é convertida para a faixa angular de operação do servomotor, sendo o valor obtido utilizado para posicionar o terceiro eixo de forma proporcional ao movimento realizado pelo usuário.

Assim como nas articulações anteriores, foram definidos identificadores específicos para o servomotor, para a variável de controle e para os pinos utilizados pelo microcontrolador, preservando a organização e a modularidade do código. Essa abordagem facilita a manutenção do programa e possibilita a expansão do sistema para novas articulações ou funcionalidades.

A Figura 13 apresenta a montagem do circuito correspondente ao terceiro eixo articulado, destacando as conexões entre o servomotor, o potenciômetro, o Arduino e a placa de prototipagem. Também são evidenciadas as ligações de alimentação, aterramento e sinal de controle, realizadas por meio de cabos de conexão, garantindo o funcionamento adequado do conjunto eletrônico.

Figura 13 – Associação de Três Servos Motores e Três Potenciômetros ao Braço Robótico



Fonte: O Autor, (2024).

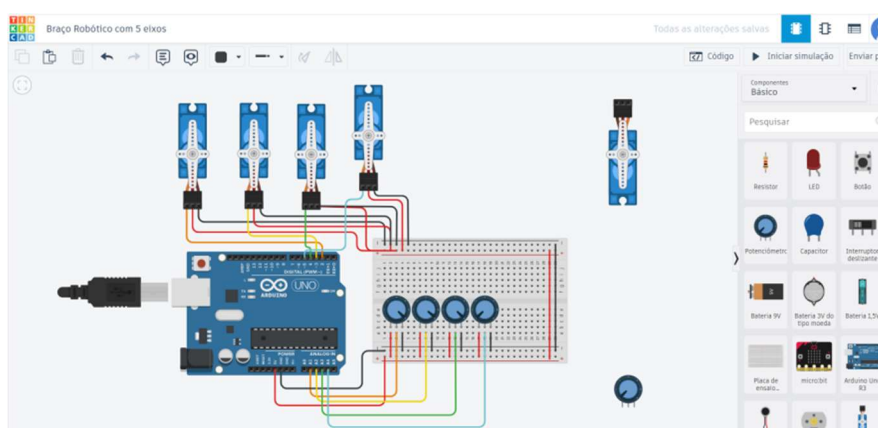
A implementação do quarto eixo articulado foi realizada seguindo a mesma metodologia adotada para os eixos anteriores. Para essa articulação, foi criado um novo objeto da biblioteca Servo, uma variável destinada ao armazenamento das leituras do quarto potenciômetro e a configuração dos respectivos pinos de entrada e saída do Arduino. Na rotina principal do programa, a leitura do potenciômetro é convertida para a faixa angular de operação do servomotor, permitindo o posicionamento do quarto eixo de acordo com o movimento realizado pelo usuário.

Mantendo a organização e a padronização do código, foram definidos identificadores específicos para o servomotor, para a variável de controle e para os pinos utilizados pelo microcontrolador. Essa estrutura modular facilita a manutenção

do programa, favorece a compreensão do código e possibilita sua ampliação para novas funcionalidades.

A Figura 14 apresenta a montagem do circuito correspondente ao quarto eixo articulado, evidenciando as conexões entre o servomotor, o potenciômetro, o Arduino e a placa de prototipagem. Também são apresentadas as ligações de alimentação, aterramento e sinal de controle, realizadas por meio de cabos de conexão, assegurando o funcionamento adequado do sistema eletrônico.

Figura 14 – Associação de Quatro Servos Motores e Quatro Potenciômetros



Fonte: O Autor, (2024).

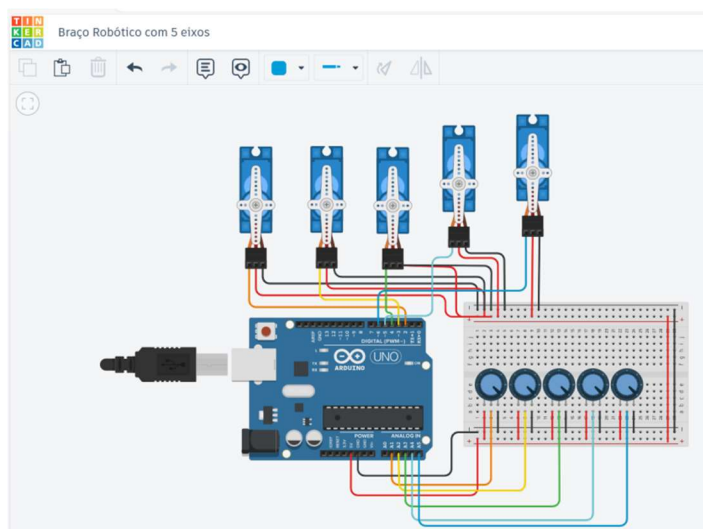
A implementação do quinto eixo articulado seguiu a mesma metodologia adotada nas etapas anteriores. Para essa articulação, foi criado um novo objeto da biblioteca Servo, bem como uma variável destinada ao armazenamento das leituras do quinto potenciômetro. Também foram configurados os respectivos pinos de entrada e saída do Arduino, mantendo a padronização da estrutura de programação utilizada em todo o projeto.

Na rotina principal do programa, a leitura do potenciômetro é convertida para a faixa angular de operação do servomotor, permitindo o controle da última articulação do braço robótico de forma proporcional ao movimento realizado pelo usuário. A organização do código em objetos, variáveis e pinos específicos para cada eixo contribui para a modularidade do sistema, facilitando sua manutenção e futuras ampliações.

A Figura 15 apresenta a montagem do circuito correspondente ao quinto eixo articulado, destacando as conexões entre o servomotor, o potenciômetro, o Arduino e

a placa de prototipagem. São evidenciadas as ligações de alimentação, aterramento e sinal de controle, realizadas por meio de cabos de conexão, completando a implementação eletrônica do braço robótico com cinco graus de liberdade.

Figura 15 – Associação de Cinco Servos Motores e Cinco Potenciômetros



Fonte: O Autor, (2024).

Após a montagem do painel de controle, composto pelos potenciômetros, e da implementação dos eixos articulados acionados pelos servomotores, foram realizados testes para verificar o funcionamento do sistema. Durante a simulação, avaliou-se a resposta de cada articulação aos comandos enviados pelos potenciômetros, permitindo validar o comportamento do braço robótico e o correto funcionamento dos componentes eletrônicos e da programação desenvolvida. Entre os eixos controlados, um foi destinado ao acionamento da garra, outro à rotação da base e os demais às articulações intermediárias do mecanismo, possibilitando a reprodução dos principais movimentos do braço robótico.

Além da visualização convencional da montagem, o Tinkercad Circuitos disponibiliza a vista esquemática do projeto, que apresenta o diagrama elétrico completo do circuito, incluindo todos os componentes e suas respectivas conexões com os pinos do Arduino. Esse recurso facilita a compreensão da arquitetura do sistema, a verificação das ligações elétricas e a reprodução do circuito em uma implementação física. O projeto completo, incluindo a montagem eletrônica e a programação desenvolvida, encontra-se disponível na plataforma Tinkercad para

consulta e reprodução: <https://www.tinkercad.com/things/cXOhmdkScqx-braco-robotico-com-5-eixos>.

Integração com o Geogebra

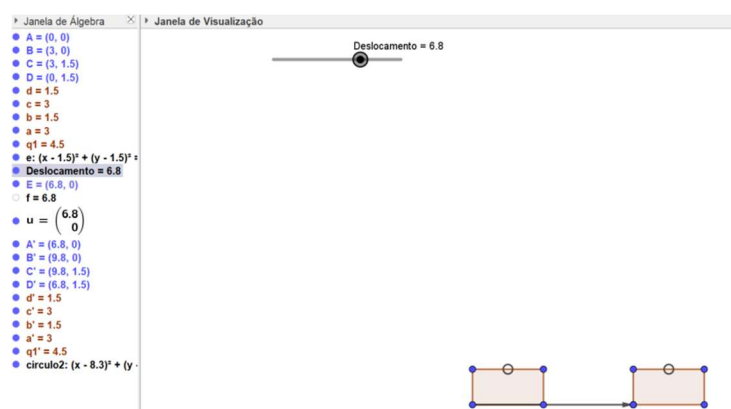
No software GeoGebra, a construção do modelo foi realizada com base em conceitos de Geometria Analítica e Geometria Plana, utilizando a janela de visualização cartesiana como referência para a definição das coordenadas dos objetos. A plataforma disponibiliza ferramentas que facilitam a criação de figuras geométricas, permitindo tanto a construção direta de polígonos quanto a definição manual de pontos para posterior formação das estruturas desejadas.

Os elementos do modelo podem ser construídos a partir da definição de coordenadas no plano, possibilitando a criação de pontos distintos que servem como base para a formação de segmentos e polígonos. Dessa forma, é possível representar as partes do sistema robótico de maneira estruturada e coerente com sua geometria.

Para representar os eixos de rotação, utilizou-se a construção de circunferências no plano cartesiano, associadas ao movimento das articulações. Esse recurso permite simular o deslocamento rotacional dos componentes por meio da relação geométrica da circunferência, definida a partir das coordenadas do centro e do raio.

Essa abordagem possibilita a modelagem geométrica do comportamento dos eixos de rotação, contribuindo para a visualização e compreensão do movimento das articulações no ambiente do GeoGebra.

Figura 16 – Deslocamento Horizontal no Geogebra



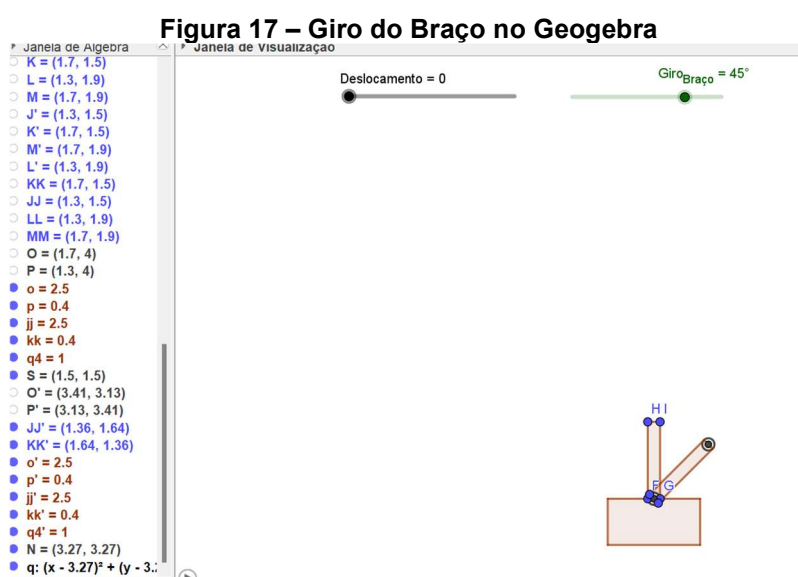
Fonte: O Autor, (2024).

Foi definido um vetor com origem no ponto inicial da base construída e extremidade associada às coordenadas controladas por um deslizante responsável pelo deslocamento. A partir desse vetor, realizou-se a translação da base e do respectivo eixo de rotação, permitindo simular o movimento de deslocamento do sistema no plano cartesiano. A Figura 16 apresenta a translação da base realizada por meio do vetor definido.

Após a implementação da movimentação da base, procedeu-se à construção do braço robótico. Inicialmente, foram estabelecidas suas coordenadas e realizada sua translação pelo mesmo princípio vetorial aplicado anteriormente. Em seguida, o braço foi submetido a uma rotação em torno do seu eixo de articulação, cujo ponto foi definido a partir do centro geométrico do sistema após a translação.

Para controlar o movimento angular, foi adicionado um segundo deslizante responsável pelo giro do braço, com variação limitada entre valores negativos e positivos, permitindo simular o movimento de rotação da articulação. Além disso, o segundo eixo de rotação foi construído com base em pontos médios, garantindo o correto posicionamento da articulação e a continuidade geométrica do modelo.

A Figura 17 apresenta o braço transladado e rotacionado em torno do eixo de rotação, evidenciando também a construção do segundo eixo articulado a partir da definição geométrica dos pontos médios.



Fonte: O Autor, (2024).

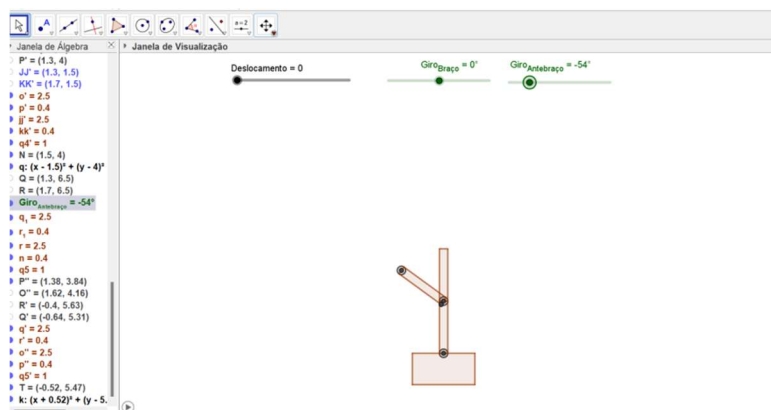
Para a construção do antebraço, foram utilizadas as coordenadas previamente definidas no ambiente do Scratch, com base na decomposição vetorial aplicada aos pontos Q e R. A partir desses pontos, foi estruturada a geometria da peça, permitindo a definição precisa de sua forma no plano cartesiano.

Em seguida, foi construído um polígono utilizando os vértices superiores do braço e os pontos Q e R, garantindo a continuidade estrutural entre as articulações do sistema. Após a modelagem inicial, o polígono foi rotacionado em torno do ponto médio de Q e R, de modo a simular o movimento do segundo eixo de rotação.

Para controlar esse movimento, foi incorporado um terceiro controle deslizante, responsável pela variação angular do antebraço. Dessa forma, o braço robótico foi progressivamente completado, incorporando mais um grau de liberdade à simulação.

A Figura 18 apresenta o antebraço já construído, bem como sua rotação em torno do segundo eixo de articulação, evidenciando o comportamento dinâmico do modelo no ambiente do GeoGebra.

Figura 18 – Giro do Antebraço no Geogebra



Fonte: O Autor, (2024).

Para a construção da garra, foram definidos pontos auxiliares no plano, permitindo a modelagem da estrutura de acordo com a forma desejada por meio da ferramenta “Polígono Rígido”. A construção teve como referência o segundo eixo de rotação, a partir do qual foi estabelecida a geometria inicial da garra.

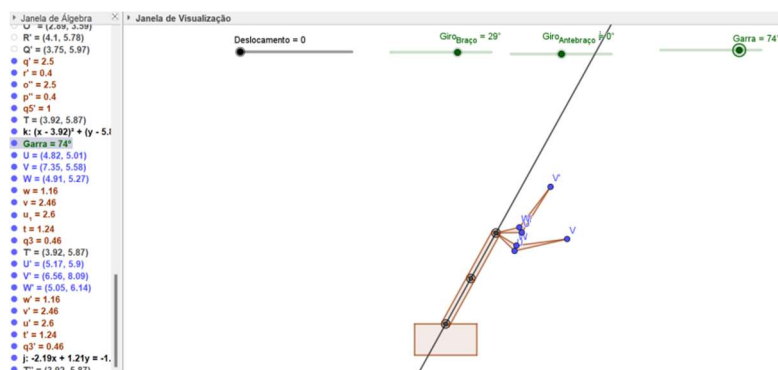
Em seguida, foi traçado um segmento de reta entre os eixos de articulação, o qual serviu como base para a construção da segunda metade da garra, garantindo simetria e coerência estrutural do modelo. Posteriormente, a garra foi submetida a

uma rotação controlada por um deslizador específico, responsável pela variação angular do movimento.

Para manter a estabilidade do sistema durante a animação, foi realizada a compensação entre os ângulos anteriores, evitando desalinhamentos na articulação e assegurando o correto comportamento dinâmico da estrutura. Dessa forma, foi possível simular de maneira consistente o movimento de abertura e fechamento da garra no ambiente do GeoGebra.

A Figura 19 apresenta a construção final da garra e a representação do eixo de articulação associado ao seu movimento.

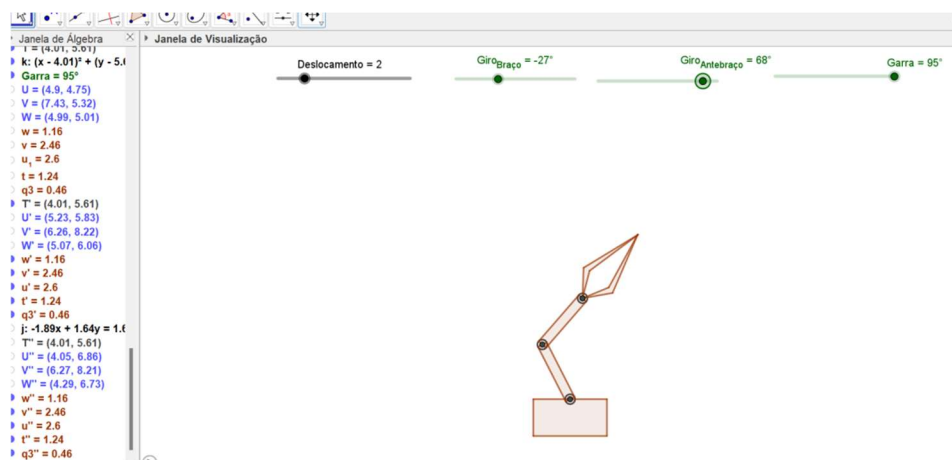
Figura 19 – Construção da Garra no Geogebra



Fonte: O Autor, (2024).

A parte da garra construída a partir da reta que liga os eixos é refletida em relação à estrutura original, permitindo a obtenção da simetria necessária para o fechamento do mecanismo. Com essa etapa, o braço robótico é concluído em sua forma integral, passando a ser totalmente manipulável por meio dos controles deslizantes implementados.

Dessa forma, o modelo final reúne todos os graus de liberdade definidos ao longo do desenvolvimento, possibilitando a simulação completa dos movimentos das articulações no ambiente do GeoGebra. A Figura 20 apresenta o braço robótico completo, evidenciando sua estrutura final e seu funcionamento dinâmico por meio dos controles interativos.

Figura 20 – Braço Robótico Completo

Fonte: O Autor, (2024).

A construção está disponível em: <https://www.geogebra.org/m/w665eaqc>.

RESULTADOS E DISCUSSÃO

Resultados Obtidos na Construção do Braço Robótico

A sequência didática proposta resultou na construção de um modelo de braço robótico em três plataformas digitais distintas — Scratch, Tinkercad e GeoGebra — permitindo analisar diferentes dimensões conceituais e operacionais envolvidas no processo de desenvolvimento. Embora a proposta ainda não tenha sido aplicada com estudantes, o desenvolvimento técnico realizado pelo docente possibilitou uma avaliação inicial das potencialidades e limitações de cada ambiente.

No Scratch, foram implementadas diferentes formas de movimentação do braço robótico, incluindo deslocamentos horizontais e verticais, rotação das articulações e controle da garra. A plataforma possibilitou a visualização clara dos movimentos articulados por meio da programação em blocos e da manipulação de elementos gráficos. A utilização de fantasias, variáveis e comandos de controle permitiu simular os eixos de rotação do sistema, favorecendo a compreensão intuitiva das transformações geométricas associadas à posição e ao movimento dos componentes.

No Tinkercad, foi desenvolvido um circuito eletrônico completo com servomotores e potenciômetros conectados a um microcontrolador Arduino Uno R3. Essa implementação permitiu simular os eixos de rotação do braço robótico em um

contexto mais próximo de um protótipo físico, possibilitando a análise do controle de movimento por meio de programação estruturada em linguagem C/C++. O ambiente contribuiu para a compreensão da relação entre entradas e saídas do sistema, bem como da necessidade de mapeamento entre sinais analógicos e digitais para o controle preciso dos ângulos de rotação.

No GeoGebra, a modelagem geométrica proporcionou a visualização dinâmica do comportamento do braço robótico, permitindo representar deslocamentos, rotações e articulações de forma integrada. O uso de controles deslizantes, vetores, translações e rotações possibilitou a explicitação dos fundamentos matemáticos envolvidos no movimento, especialmente conceitos de Geometria Plana e Analítica que, nas demais plataformas, permanecem implícitos. Além disso, a ferramenta favoreceu maior agilidade na construção dos modelos, embora tenha exigido maior domínio conceitual matemático para sua implementação.

De forma integrada, os resultados evidenciam a viabilidade técnica da proposta e demonstram que a articulação entre diferentes tecnologias digitais pode contribuir significativamente para o ensino de matemática e robótica, promovendo uma abordagem interdisciplinar e contextualizada dos conteúdos.

Limitações Identificadas

A principal limitação refere-se à ausência de aplicação empírica da sequência didática com estudantes, o que impossibilita a avaliação de sua eficácia pedagógica em contextos reais de ensino e aprendizagem.

Outro aspecto relevante é a dependência de infraestrutura tecnológica, uma vez que a utilização integrada das três plataformas requer computadores com acesso à internet e recursos adequados, o que pode restringir sua implementação em instituições com baixa disponibilidade tecnológica.

Observam-se ainda limitações inerentes às próprias ferramentas utilizadas: o Scratch apresenta restrições por não operar plenamente em ambiente tridimensional, o que limita a representação de certos movimentos rotacionais; o Tinkercad, embora adequado para simulação eletrônica, não permite a visualização física integrada do comportamento mecânico do sistema; e o GeoGebra, apesar de sua potência para

modelagem geométrica, pode representar uma barreira inicial devido à exigência de maior domínio matemático.

Adicionalmente, não foram realizadas estimativas de custos para a eventual construção física do braço robótico com Arduino e servomotores, o que limita a análise de viabilidade financeira da proposta.

Por fim, destaca-se a ausência de validação pedagógica, uma vez que não há dados empíricos sobre a resposta dos estudantes à proposta ou sobre seus impactos no desempenho em Matemática e no desenvolvimento do pensamento computacional.

O reconhecimento dessas limitações é fundamental para orientar pesquisas futuras e fortalecer a consistência e a credibilidade do estudo.

Comparação com Trabalhos Semelhantes

A literatura apresenta diversas iniciativas relacionadas à construção e ao uso de braços robóticos com fins educacionais, especialmente no contexto do ensino de robótica, programação e engenharia. Em geral, esses estudos exploram o uso de plataformas isoladas ou protótipos físicos baseados em microcontroladores, com destaque para o uso do Arduino e de simuladores eletrônicos como ferramentas de apoio ao ensino (Kato, 2015; Jahnavi & Sivraj, 2017).

Kato (2015) discute o desenvolvimento de protótipos físicos aplicados ao ensino de Engenharia, enfatizando a articulação entre conceitos mecânicos e eletrônicos na construção de sistemas robóticos educacionais. O autor destaca a relevância do aprendizado baseado em projetos como estratégia para aproximar teoria e prática, embora a abordagem permaneça centrada em ambientes físicos de prototipagem.

Na mesma linha, Jahnavi e Sivraj (2017) apresentam modelos educacionais voltados ao ensino de cinemática por meio de simulações computacionais, evidenciando o potencial dessas ferramentas para a visualização de movimentos e para a compreensão de conceitos abstratos. Entretanto, os autores concentram-se em ambientes únicos de simulação, sem integração com outras plataformas de desenvolvimento ou com representações geométricas formais.

Outros estudos também reforçam o uso do Arduino como ferramenta central em projetos de robótica educacional. Maloney et al. (2010) destacam o potencial de ambientes de programação em blocos para introdução de conceitos de computação, enquanto Mataric (2007) enfatiza o uso da robótica como meio de engajamento em disciplinas de ciência, tecnologia, engenharia e matemática (STEM). Ainda assim, essas abordagens tendem a priorizar um único ambiente de desenvolvimento, seja físico ou virtual, sem integração entre diferentes níveis de representação.

De forma mais recente, estudos como os de Grover e Pea (2013) discutem o desenvolvimento do pensamento computacional por meio de atividades estruturadas em ambientes de programação visual, reforçando a importância de ferramentas como Scratch no ensino básico. No entanto, tais abordagens geralmente não articulam explicitamente os conceitos matemáticos subjacentes às transformações geométricas envolvidas nos movimentos dos sistemas robóticos.

Nesse contexto, o presente trabalho diferencia-se por propor uma abordagem integrada entre três ambientes computacionais complementares: Scratch, Tinkercad e GeoGebra. Enquanto o Scratch possibilita a visualização intuitiva por meio de programação em blocos, o Tinkercad permite a simulação eletrônica de circuitos com Arduino, e o GeoGebra viabiliza a formalização matemática dos movimentos por meio de Geometria Analítica e transformações geométricas.

Essa integração possibilita uma abordagem multimodal do ensino de robótica, na qual diferentes níveis de abstração são articulados: o nível visual e algorítmico (Scratch), o nível físico-eletrônico (Tinkercad) e o nível matemático-formal (GeoGebra). Tal convergência amplia as possibilidades pedagógicas, favorecendo a compreensão conceitual e a interdisciplinaridade.

Assim, ao contrário da maioria dos trabalhos encontrados na literatura, que privilegiam uma única ferramenta ou perspectiva, a proposta deste estudo se destaca por integrar múltiplas representações do mesmo sistema robótico, contribuindo para uma visão mais ampla e estruturada do processo de modelagem, simulação e controle de braços robóticos em contextos educacionais.

CONCLUSÕES E PERSPECTIVAS FUTURAS

O desenvolvimento da sequência didática voltada à construção de um braço robótico por meio das plataformas Scratch, Tinkercad e GeoGebra evidenciou o potencial das tecnologias digitais como recursos pedagógicos inovadores para o ensino de Matemática. A integração entre diferentes ambientes computacionais permitiu articular distintas formas de representação — algorítmica, eletrônica e geométrica — favorecendo uma abordagem interdisciplinar e alinhada às demandas contemporâneas da educação.

Os resultados obtidos indicam que cada plataforma contribui de maneira complementar para a compreensão do sistema robótico. O Scratch favorece a exploração intuitiva de algoritmos, coordenadas e eventos; o Tinkercad aproxima o estudante da lógica de circuitos eletrônicos e do controle de atuadores por meio do Arduino; e o GeoGebra possibilita a visualização e formalização de conceitos matemáticos, como transformações geométricas, vetores e relações trigonométricas. Essa convergência amplia as possibilidades de aprendizagem ao conectar diferentes níveis de abstração em um mesmo objeto de estudo.

A proposta também se mostra coerente com abordagens teóricas que defendem a aprendizagem ativa, a construção do conhecimento e o uso de tecnologias digitais na educação, especialmente no contexto da robótica educacional e do pensamento computacional. Nesse sentido, o braço robótico construído virtualmente configura-se como um objeto de aprendizagem que integra programação, matemática e engenharia de forma contextualizada.

Apesar dos avanços apresentados, reconhecem-se limitações importantes, sobretudo a ausência de aplicação empírica com estudantes, o que impossibilita a avaliação dos impactos reais da proposta no processo de ensino e aprendizagem. Soma-se a isso a dependência de infraestrutura tecnológica adequada e a necessidade de formação docente para o uso integrado das plataformas utilizadas.

Como perspectivas futuras, destaca-se a necessidade de aplicação da sequência didática em contextos reais de sala de aula, com coleta e análise de dados qualitativos e quantitativos sobre a aprendizagem dos estudantes. Além disso, pretende-se avançar para a construção de um protótipo físico do braço robótico com Arduino, permitindo validar a compatibilidade entre os modelos virtuais e físicos.

Outras possibilidades incluem a ampliação dos modelos em Scratch e GeoGebra para representar movimentos mais próximos da cinemática tridimensional, bem como a investigação do impacto da proposta no ensino de conteúdos matemáticos específicos, como funções trigonométricas, vetores, ângulos e transformações geométricas. Também se destaca a importância do desenvolvimento de materiais de apoio ao professor, como roteiros didáticos e tutoriais, que possam facilitar a replicação da proposta em diferentes contextos educacionais.

Conclui-se que a integração entre robótica educacional e tecnologias digitais diversas constitui uma estratégia promissora para o ensino de Matemática, contribuindo para uma aprendizagem mais significativa, contextualizada e alinhada às exigências da cultura digital contemporânea, ao mesmo tempo em que abre caminho para novas investigações e aplicações pedagógicas.

REFERÊNCIAS

ABRANTES, A., SERRAZINA, L., e OLIVEIRA, I. **A Matemática na Educação Básica**. Lisboa: Ministério da Educação – Departamento da Educação Básica. 1999.
BLIKSTEIN, Paulo. **O pensamento computacional e a reinvenção do computador na educação**. Education & Courses, v. 1. 2008;

BORBA, M. de C.; PENTEADO, M. G. **Informática e Educação Matemática**. 5ª ed. Belo Horizonte: Autêntica Editora. 2012.

BORDINI, et al. **Computação na educação básica no Brasil: estado da arte**. Rev. Inform. Teor. Apl. (Online), Porto Alegre, v. 23, n. 2, p. 210-238, novembro. 2016;
BRACKMANN, C. P. **Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica**. Tese (Doutorado) — UFRGS, Porto Alegre, 2017. Disponível em: lume.ufrgs.br/bitstream/handle/10183/172208/001054290.pdf?sequence=1&isAllowed=y. Acesso em 28 jul. 2024.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular**. Brasília, 2018. Disponível em: http://basenacionalcomum.mec.gov.br/images/BNCC_20dez_site.pdf. Acesso em: 02 ago. 2024.

BROWN, Tim. **Design Thinking**. *Harvard Business Review*, v. 86, n.6, p. 84-92, 141, 2008. Disponível em: <https://readings.design/PDF/Tim%20Brown,%20Design%20Thinking.pdf>. Acesso em 19 jul. 2024.

CÂMARA, J., MACHADO, A., WILLIAMS, V. (2018). **Robótica Educacional: Desenvolvendo Competências para o Século XXI**. *Cultura Maker na Escola*.

Disponível em: http://ceur-ws.org/Vol-2185/CtrlE_2018_paper_50.pdf, p. 215–226.
Acesso em: 03 ago. 2024.

CAMPOS, F. R. **A robótica para uso educacional**. São Paulo: Editora Senac. 2019.
DE MEDEIROS, Luciano Frontino; WÜNSCH, Luana Priscila. **Ensino de programação em robótica com Arduino para alunos do ensino fundamental: relato de experiência**. Revista Espaço Pedagógico, v. 26, n. 2, p. 456-480. 2019.

FERNANDES, Carla da Costa. **Seduc: Um simulador de ambiente de robótica educacional em plataforma virtual**. Dissertação de Mestrado. Universidade Federal do Rio Grande do Norte. 2013.

GROVER, Shuchi; PEA, Roy. **Computational Thinking in K–12: A Review of the State of the Field**. *Educational Researcher*, v. 42, n. 1, p. 38–43, 2013.

JAHNAVI, Kadirimangalam; SIVRAJ, P. **Teaching and Learning Robotic Arm Model**. In: 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT). IEEE. p 1570-1575. 2017.

KATO, Lucas Kenzo. **Projeto de um braço robótico para fins didáticos**. UFSC. Florianópolis. 2015. Disponível em: [//core.ac.uk/download/pdf/78552479.pdf](http://core.ac.uk/download/pdf/78552479.pdf).
Acesso em: 12 ago. 2024.

MALONEY, John; RESNICK, Mitchel; RUSK, Natalie; SILVERMAN, Brian; EASTMOND, Evelyn. **The Scratch Programming Language and Environment**. *ACM Transactions on Computing Education*, v. 10, n. 4, art. 16, p. 1–15, 2010.

MARCHETTI, Josiane Mazurana; KLAUS, Vanessa Lucena Camargo de Almeida. **Software GeoGebra: um recurso interativo e dinâmico para o ensino de Geometria Plana**. Caderno PDE, Curitiba, v. I. 2014.

MATARIĆ, Maja J. **The Robotics Primer**. Cambridge: MIT Press, 2007.

PACHECO, Erica Farias. **Utilizando o software GeoGebra no ensino da Matemática: uma ferramenta para construção de gráficos de parábolas e elipses no 3º ano do Ensino Médio**. *Debates em Educação*, Maceió, v. 11, nº 24, Maio/Ago. 2019.

PAPERT, S. **A máquina das crianças: repensando a escola na era da informática**. Edição Revisada. Artmed, Porto Alegre. 2008.

PENTEADO, Miriam; BORBA, Marcelo C. **A informática em ação: formação de professores, pesquisa e extensão**. São Paulo: Olho D'Água. 2000.

PEREIRA, Giancarlo Secci. **A linguagem de programação educativa Scratch na produção de conteúdos digitais para mediação da aprendizagem de Matemática na educação básica**. 126 p. Dissertação (Mestrado Profissional em Matemática) - Programa de Pós-Graduação em Matemática em Rede Nacional, Universidade Federal do Pará, Belém. 2019.

PLATTNER, Hasso et al. **Design thinking: integrating innovation, customer experience, and brand value**. Springer Science & Business Media. 2011.

SILVA, Manoel Roberto Alves da. **A utilização do software Geogebra no processo de ensino-aprendizagem da Geometria plana**. Universidade Federal de Alagoas. PROFMAT. Maceió: 2017. Disponível em: <https://www.repositorio.ufal.br/jspui/bitstream/riufal/1756/1/A%20utiliza%C3%A7%C3%A3o%20do%20software%20Geogebra%20no%20processo%20de%20ensino-aprendizagem%20da%20Geometria%20plana.pdf>. Acesso em: 10 ago.2024.

WING, J. M. **Computational thinking**. *Communications of the ACM*, v. 49, n. 3, p. 33. 2006.

YOUNG DIGITAL PLANET S.A. et al. **Educação no século 21: Tendências, ferramentas e projetos para inspirar**. Trad. Danielle Mendes Sales. São Paulo: Fundação Santillana. 2016.

AUTORES

Adenir Dos Santos Camargo, possui Mestrado Profissional em Matemática Pura e Aplicada - PROFMAT, pela Universidade Tecnológica e Federal do Paraná - UTFPR, Campus Pato Branco, Especialização em Ensino de Matemática pela Faculdade Educacional da Lapa - FAEL, Especialização em Educação a Distância e as Tecnologias Educacionais e Especialização em Tecnologias para Aplicações Web pela Universidade do Norte do Paraná - UNOPAR, Especialização em Robótica Educacional e suas Tecnologias no Ensino de Matemática pela Universidade Federal de Catalão - UFCAT, Campus Jataí, Licenciatura Plena em Matemática pela Universidade Alto Vale do Rio do Peixe - UNIARP, Licenciatura em Física pela Universidade Metropolitana do Estado de São Paulo - UNIMES, Licenciatura em Informática pela UDESC. Atualmente é Professor Substituto da Universidade do Estado de Santa Catarina - UDESC no Centro de Educação Superior do Meio Oeste - CESMO, onde ministra as disciplinas de Matemática Básica, Arquitetura de Computadores, Matemática Discreta, Cálculos Matemáticos Aplicados a Administração Pública e Métodos Estatísticos para os Cursos de Sistemas de Informação e Administração Pública, Professor Efetivo da Secretaria Estadual de Educação na Cidade de Caçador/SC na Escola de Ensino Fundamental Graciosa Copetti Pereira, professor do Ensino Fundamental e Médio do Colégio Bom Jesus Aurora em Caçador/SC, professor voluntário do Curso de Geogebra, OGEOGBRA e multiplicador do Núcleo de Tecnologias Educacionais - NTE da 10 Coordenadoria Regional de Educação - CRE de Caçador/SC. E-mail: professoradenirsenac@gmail.com.

Flavia Gonçalves Fernandes, possui graduação em Biomedicina pelo Centro Universitário São Lucas (2016) e foi bolsista de iniciação científica por 3 anos no

Laboratório de Epidemiologia Genética da Fundação Oswaldo Cruz FIOCRUZ. Possui mestrado em Oncologia Molecular pelo Hospital de Câncer de Barretos e experiência na área de Genética, Bioinformática, Oncologia e Pesquisa Clínica. Foi Coordenadora de Projetos do Biobanco do Hospital de Câncer de Barretos, onde desempenhou ações voltadas para inclusão de pacientes na pesquisa. Atualmente é Coordenadora de Estudos Clínicos na Unidade de Pesquisa Clínica do Hospital de Câncer de Barretos, dando suporte na condução de diversos estudos com seres humanos, auxiliando ações entre a indústria farmacêutica, o investigador e os participantes de pesquisa. E-mail: flavia.fernandes92@gmail.com.