

JEANNE MOREIRA DE SOUSA
MARCUS VINICIUS MALTEMPI



ROBÓTICA EDUCACIONAL COM METODOLOGIAS ATIVAS

FORMAÇÃO INICIAL E CONTINUADA
DE PROFESSORES



Esta obra é licenciada sob uma licença **Creative Commons Atribuição-NãoComercial-SemDerivações (CC BY-NC-ND 4.0)**. Esta licença permite a cópia e redistribuição do material, em qualquer suporte ou formato, sem fins comerciais, desde que o crédito seja atribuído ao autor. Traduções, adaptações ou outras modificações a partir do material não podem ser distribuídas. Os termos da licença são detalhados em <https://creativecommons.org/licenses/by-nc-nd/4.0/legalcode.pt>.

Dados Internacionais de Catalogação na Publicação (CIP)

Sousa, Jeanne Moreira de

Robótica educacional com metodologias ativas: formação inicial e continuada de professores [recurso eletrônico] / Jeanne Moreira de Sousa e Marcus Vinicius Maltempi. — Rio Claro : UNESP - Instituto de Geociências e Ciências Exatas, 2025.

178 p. : il. color., fotos color. ; pdf.

1. Robótica educacional 2. Programação em blocos 3. Metodologias ativas 4. Formação de professores 5. Pensamento computacional

DESCRIÇÃO TÉCNICA

Origem do Produto: Desenvolvido no âmbito da pesquisa de pós-doutorado intitulada “Uso da ABP por meio de Softwares Educacionais para o desenvolvimento do Pensamento Computacional na EPT”.

Área de Concentração: Educação Matemática.

Linha de Pesquisa: Matemática e o Pensamento Computacional nos Processos de Ensino e Aprendizagem de Matemática.

Público-Alvo: Docentes de Matemática dos Ensinos Fundamental, Médio e da Educação Profissional e Tecnológica (EPT).

Finalidade: Apresentar as atividades desenvolvidas no curso de formação docente e oferecer recomendações práticas para sala de aula de Matemática.

Categoria do Produto: Manual com orientações.

Estruturação do Produto: Em capítulos.

Ilustrações: Para as ilustrações e o layout deste produto foram utilizados os recursos e elementos gráficos da plataforma Canva.

Imagens: Para a divulgação das imagens dos cursistas foram dadas as devidas autorizações.

Registro do Produto/Ano: 2025.

Disponibilidade: Irrestrita.

Divulgação: Por meio digital.

Avaliação: Seminário de Matemática e Educação Matemática (SMEM) do Programa de Pós-Graduação em Educação Matemática (PPGEM) da UNESP - *Campus* Rio Claro.

Instituições envolvidas: Instituto Federal do Amazonas (IFAM), Universidade Estadual Paulista (UNESP) e Fundação de Amparo à Pesquisa do Estado do Amazonas (FAPEAM).

Idioma: Português.

Cidade: Manaus.

País: Brasil.



RESUMO DO PRODUTO EDUCACIONAL

A crescente demanda por habilidades do século XXI – cognitivas, socioemocionais e tecnológicas – impõe desafios significativos à formação de professores, especialmente na área de matemática. A Base Nacional Comum Curricular (BNCC), documento normativo da educação escolar brasileira, preconiza a integração dessas habilidades no cotidiano escolar, mas educadores carecem de subsídios para sua efetiva transposição para a prática. Este trabalho descreve o desenvolvimento e análise de um curso de formação docente semipresencial, intitulado "Robótica Educacional com Metodologias Ativas", para professores de matemática do Ensino Fundamental (EF) e Médio (EM) e da Educação Profissional e Tecnológica (EPT). O curso foi oferecido no contexto de uma pesquisa de pós-doutorado em Educação Matemática realizada na Universidade Estadual Paulista (UNESP) - *Campus* de Rio Claro. O objetivo principal foi orientar os docentes a integrarem a robótica educacional e metodologias ativas, como a Aprendizagem Baseada em Projetos (ABP) e a Rotação por Estações (RE), ao ensino de matemática. A metodologia promoveu uma abordagem "mão na massa", utilizando placas embarcadas (Arduino Uno, Micro:bit e Makey Makey), programadas a partir do software de programação em blocos Pictoblox, no desenvolvimento de projetos práticos que conectassem conceitos matemáticos e Pensamento Computacional (PC). Os projetos, aplicáveis aos EF/EM, foram organizados em torno de dois Temas Gerais: "Cidades Inteligentes" (controle de iluminação, controle de tráfego e sistemas de alarmes) e "Sensores e Dados para um Ambiente Sustentável" (monitoramento ambiental e reciclagem).

As atividades exploraram conceitos como aritmética, razão, proporção, lógica booleana, sequências, otimização, gráficos, grandezas, estatística, contagem, probabilidade e classificação.



O curso tornou compreensíveis tecnologias e metodologias, tornando-as acessíveis e aplicáveis, promovendo a percepção de competências da BNCC tais como pensamento crítico, cultura digital, comunicação, trabalho em equipe e responsabilidade cidadã, além de habilidades matemáticas em resolução de problemas, análise de dados e modelagem. A avaliação dos questionários de pesquisa, aplicados aos docentes, foi qualitativa, o que nos levou a compreender suas expectativas, percepções e experiências. Por exemplo, a avaliação diagnóstica inicial revelou conhecimento limitado em PC, ABP e RE. Também, ninguém possuía experiência em eletrônica básica (essencial para montagem segura de circuitos), mas pela simplicidade dos projetos propostos, essa limitação foi minimizada. Suas expectativas eram focadas na aquisição de conhecimentos práticos para aplicabilidade em sala de aula. A avaliação final indicou aumento significativo na compreensão e confiança em PC e metodologias ativas (ABP e RE), demonstrando assim, a eficácia da abordagem experiencial. Embora a interação com o software de programação tenha sido variada, sugestões apontaram para a necessidade de aprofundamento e de uso recorrente, igualmente a manipulação de circuitos eletrônicos em plataformas online, além do desejo por mais encontros práticos. Em síntese, o curso transformou a percepção e habilidades dos professores, permitindo-lhes implementar a robótica educacional como ferramenta para o ensino da matemática. Os resultados reforçam a importância de formações continuadas que abordem tecnologias digitais para a Educação de forma contextualizada e prática, capacitando educadores para a formação integral dos estudantes.

É nessa perspectiva que este material, fruto de um trabalho intenso e gratificante, chega às mãos de todos os professores interessados em promover um ensino de matemática efetivo e de qualidade.





SOBRE OS AUTORES

Jeanne Sousa possui licenciatura, bacharelado e mestrado em Matemática pela UFAM, doutorado em Clima e Ambiente pelo INPA/UEA, pós-doutorado em Modelagem Numérica e Previsão de Tempo e Clima no LMC/INPA. Atualmente é professora de Matemática no IFAM e aluna de pós-doutorado do Programa de Pós-graduação em Educação Matemática da UNESP.



Marcus Maltempi é graduado em Ciência da Computação pela UNESP, mestre em Ciências da Computação e Matemática Computacional pela USP, doutor em Engenharia Elétrica e de Computação pela Unicamp (sanduíche na Universidade de Toronto, Canadá), além de pós-doutorado na Universidade de Londres (Inglaterra). Atualmente, é Livre-docente em Educação Matemática pela UNESP e Professor Associado III da Unesp de Rio Claro.



Também é membro do Grupo de Pesquisa Diálogos e Indagações sobre Escolas e Educação Matemática (DIEEM) e da SBEM (Sociedade Brasileira de Educação Matemática), além de bolsista Produtividade em Pesquisa do CNPq.





SUMÁRIO

| | |
|--|-----------|
| Introdução | 13 |
| Capítulo 1 | 17 |
| BASE PEDAGÓGICA - APRENDIZAGEM ATIVA NA PRÁTICA | |
| A BASE NACIONAL COMUM CURRICULAR E AS HABILIDADES PARA O SÉCULO XXI | 18 |
| PENSAMENTO COMPUTACIONAL: INTER-RELAÇÕES COM O APRENDIZADO DA MATEMÁTICA | 21 |
| CULTURA MAKER E CONSTRUCIONISMO: ALINHAMENTO COM AS COMPETÊNCIAS DA BNCC | 25 |
| APRENDIZAGEM BASEADA EM PROJETOS: MATEMÁTICA COM PROPÓSITO PARA A BNCC | 28 |
| ROTAÇÃO POR ESTAÇÕES: ORGANIZANDO A SALA DE AULA PARA A EXPERIMENTAÇÃO E O DESENVOLVIMENTO DE COMPETÊNCIA | 33 |
| ROBÓTICA EDUCACIONAL: CONSTRUINDO E PROGRAMANDO A APRENDIZAGEM | 37 |
| Capítulo 2 | 41 |
| CONHECENDO AS FERRAMENTAS | |
| PICTOBLOX: A PORTA DE ENTRADA PARA A PROGRAMAÇÃO | 42 |
| ARDUINO: VERSATILIDADE E POTENCIAL | 50 |
| MICRO:BIT: SIMPLES E PRÁTICA | 57 |

| | |
|---|------------|
| O MAKEY MAKEY: TRANSFORMANDO O COTIDIANO EM INTERFACE | 63 |
| Capítulo 3 | 69 |
| CURSO DE FORMAÇÃO DOCENTE | |
| ENCONTRO 1: “OLÁ MUNDO!” | 70 |
| PROJETO 1.1: ACENDENDO UM LED (“OLÁ MUNDO!”) | 72 |
| ENCONTRO 2: CIDADES INTELIGENTES | 80 |
| PROJETO 2.1: CONTROLE DE ILUMINAÇÃO E SEGURANÇA | 80 |
| PROJETO 2.2: CONTROLE DE ILUMINAÇÃO COM SENSOR DE PRESENÇA (PIR) | 88 |
| PROJETO 2.3: CONTROLE DE TRÁFEGO | 95 |
| PROJETO 2.4: SISTEMA DE ALARME | 104 |
| ENCONTRO 3: SENSORES E DADOS PARA UM AMBIENTE SUSTENTÁVEL | 113 |
| PROJETO 3.1: MONITORAMENTO DE TEMPERATURA E HUMIDADE | 113 |
| PROJETO 3.2: MONITORAMENTO DE GASES INFLAMÁVEIS | 121 |
| PROJETO 3.3: SENSOR DE LUZ E TERMÔMETRO | 128 |
| PROJETO 3.4: REGISTRADOR DE DADOS | 136 |
| PROJETO 3.5: JOGO DA RECICLAGEM | 141 |
| Capítulo 4 | 153 |
| A VOZ DA EXPERIÊNCIA - ANÁLISE DO CURSO DE FORMAÇÃO METODOLOGIA DE ANÁLISE DE DADOS: UMA ABORDAGEM QUALITATIVA | 154 |

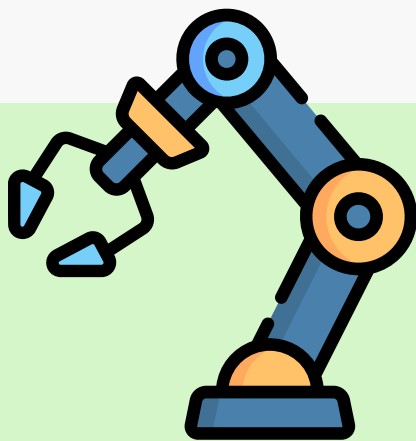
| | |
|---|------------|
| PERFIL PROFISSIONAL E EXPERIÊNCIA PRÉVIA DOS PARTICIPANTES | 156 |
| AVALIAÇÃO DIAGNÓSTICA: UM PANÓRAMA INICIAL | 158 |
| AVALIAÇÃO DE SATISFAÇÃO: O IMPACTO DO CURSO | 163 |
| A TRAJETÓRIA DO APRENDIZADO: DO DIAGNÓSTICO À SATISFAÇÃO FINAL | 169 |
| Referências | 173 |
| Agradecimentos | 176 |



INTRODUÇÃO

No cenário educacional contemporâneo, a crescente demanda por habilidades do século XXI, classificadas como cognitivas, socioemocionais e tecnológicas, impõe novos e complexos desafios à formação docente. A BNCC, Base Nacional Comum Curricular (Brasil, 2017) preconiza a integração dessas competências no ambiente escolar, estendendo-se ao raciocínio lógico, a decomposição de problemas, o reconhecimento de padrões, a abstração e a formulação de algoritmos, pilares do Pensamento Computacional (Wing, 2006; 2011).

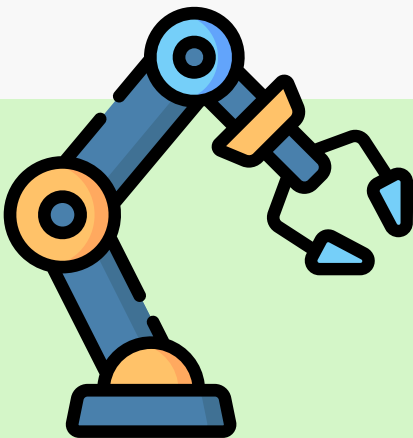
No contexto da matemática, essas habilidades são cruciais para capacitar os alunos a resolver problemas de maneira eficiente e estruturada, aplicáveis em diversas áreas do conhecimento (Wing, 2011). No entanto, um dos principais desafios para a implementação eficaz do pensamento computacional nas escolas de educação básica reside na formação do professor. Muitos educadores, conforme apontado por Valente (2016), ainda carecem das habilidades necessárias para utilizar ferramentas tecnológicas como softwares de programação e placas embarcadas em suas aulas.



Essa lacuna demanda propostas de capacitações específicas, essenciais para que os professores possam integrar essas tecnologias ao currículo e às suas práticas pedagógicas de maneira eficiente, promovendo uma abordagem ativa e interdisciplinar.

A utilização de ferramentas digitais, como softwares de programação, também tem se mostrado fundamental para estimular um maior entendimento de conceitos matemáticos por meio da experimentação e da prática, proporcionando uma aprendizagem significativa (Sápiras, Vecchia e Maltempi, 2015; Riboldi e Reichert, 2020).

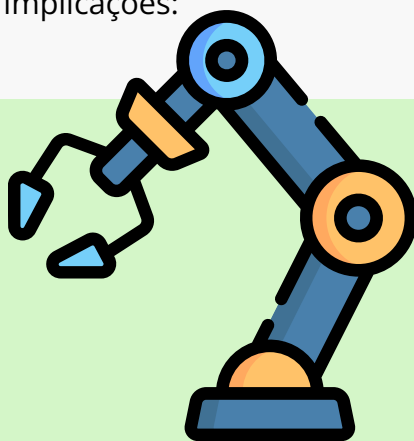
Nesse contexto, este material apresenta a proposta e os resultados de um curso de formação inicial e continuada de professores de matemática, que os instruiu para integrarem a robótica educacional e as metodologias ativas em suas práticas de sala de aula. O curso baseou-se na teoria construcionista de Seymour Papert (1994), que, expandindo o construtivismo de Jean Piaget (Piaget e Greco, 1974), defende que o aprendizado ocorre de forma mais eficaz quando o aluno está envolvido na criação de artefatos significativos, como programas de computador ou protótipos tecnológicos.



Essa abordagem "mão na massa" foi aplicada utilizando softwares educacionais como o Pictoblox e as placas embarcadas Arduino Uno, Micro:bit e Makey Makey. As metodologias ativas empregadas, como a Aprendizagem Baseada em Projetos (ABP) e a Rotação por Estações (RE), buscaram uma abordagem prática, interdisciplinar e colaborativa para a solução de problemas do dia a dia, ao mesmo tempo em que estimulavam a percepção e o desenvolvimento dessas habilidades (Bender, 2014; Horn, Staker e Christensen, 2015; Bacich e Moran, 2018).

A influência das Tecnologias Digitais (TDs) em nossas vidas é inegável e, como defende Maltempo (2008), é imperativo que a sala de aula também reflita essa transformação, o que requer do professor um repensar de sua prática. Assim, além de desenvolver as habilidades técnicas necessárias para utilizar essas ferramentas, o curso também enfatizou a importância de formar professores que possam atuar como mediadores do conhecimento, guiando os alunos no processo de experimentação e construção ativa de seu próprio conhecimento (Papert, 1994; Valente, 2016).

Este trabalho está estruturado para guiar o leitor por essa jornada de formação e suas implicações:



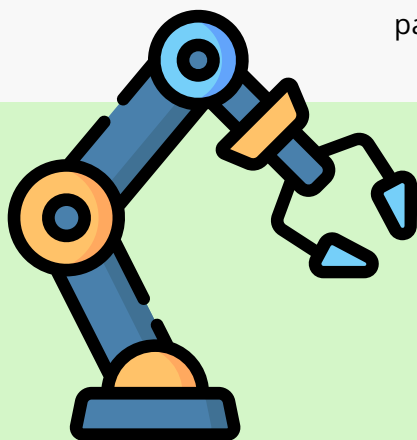
Capítulo 1 - A Base Pedagógica - Aprendizagem Ativa na Prática: aprofunda os fundamentos teóricos que sustentaram o curso, desde as competências da BNCC até as metodologias ativas e o pensamento computacional.

Capítulo 2 - Conhecendo as Ferramentas: apresenta detalhadamente o software de Programação em Blocos Pictoblox, e as placas embarcadas Arduino Uno, Micro:bit e Makey Makey utilizadas.

Capítulo 3 - Mãos na Massa - Projetos Guiados para as Aulas de Matemática: descreve passo a passo os projetos práticos desenvolvidos no curso, destacando sua aplicação em conceitos matemáticos e o alinhamento com a BNCC.

Capítulo 4 - A Voz da Experiência - Análise do Curso de Formação: apresenta a metodologia de análise e os resultados das avaliações diagnóstica e de satisfação dos professores, fornecendo insights sobre o impacto e as sugestões de aprimoramento do curso.

Com esta abordagem, esperamos que o professor leitor possa não apenas dominar as ferramentas e metodologias aqui apresentadas, mas também se sinta inspirado a propor projetos educacionais inovadores, tornando o aprendizado da matemática mais envolvente, dinâmico, criativo e relevante para os desafios do mundo digital.



CAPÍTULO 1

BASE PEDAGÓGICA - APRENDIZAGEM ATIVA NA PRÁTICA



A BASE NACIONAL COMUM CURRICULAR E AS HABILIDADES PARA O SÉCULO XXI

A BNCC, Base Nacional Comum Curricular (Brasil, 2017), documento normativo que define o conjunto orgânico e progressivo de aprendizagens essenciais que todos os alunos da Educação Básica brasileira devem desenvolver é a base de nossa proposta.

Referência obrigatória para a elaboração e adequação dos currículos e propostas pedagógicas nas redes de ensino público e privado, também é uma referência para a formação de professores e para a elaboração de conteúdos educacionais.

Ela vai além da mera listagem de conteúdos, focando no desenvolvimento de competências e habilidades que preparem os estudantes para os desafios do mundo contemporâneo, para que eles sejam capazes de compreender, interagir e inovar em um mundo totalmente digitalizado, que assegure uma formação humana integral e que vise à construção de uma sociedade justa, democrática e inclusiva.



Na BNCC, competência é definida como a mobilização de conhecimentos (conceitos e procedimentos), habilidades (práticas, cognitivas e socioemocionais), atitudes e valores para resolver demandas complexas da vida cotidiana, do pleno exercício da cidadania e do mundo do trabalho (Brasil, 2017).

Em síntese:

1. Conhecimento: oportuniza a compreensão de conceitos científicos, tecnológicos e matemáticos (Grandezas, Medidas, Geometria, Probabilidade, Estatística, Álgebra e Números).

2. Pensamento Científico, Crítico e Criativo: estimula a investigação, a formulação de hipóteses, a análise de dados e a resolução de problemas complexos, habilidades intrínsecas ao Pensamento Computacional e ao "fazer" da Cultura Maker.

3. Repertório Cultural: permite a exploração de diversas formas de expressão, incluindo as digitais e artísticas envolvidas na criação de protótipos.

4. Comunicação: desenvolve a capacidade de expressar ideias, interpretar informações e interagir em diferentes linguagens (verbal, visual, digital), essencial para a apresentação de projetos.

5. Cultura Digital: promove o uso crítico, significativo e ético das tecnologias digitais para resolver problemas, comunicar e produzir conhecimento, integrando hardware e software.

6. Trabalho e Projeto de Vida: incentiva a proatividade, a colaboração e a organização, importantes para a execução de projetos e a visão de futuro.

7. Argumentação: desenvolve a capacidade de formular, negociar e defender ideias com base em dados e evidências, comum na depuração de códigos e no compartilhamento de resultados.

8. Autoconhecimento e Autocuidado: contribui para a compreensão das emoções e a gestão do tempo e do esforço no processo de aprendizagem e criação.

9. Empatia e Cooperação: fortalece o trabalho em equipe, o respeito às diferenças e a construção de soluções coletivas, pilares da metodologia de Rotação por Estações.

10. Responsabilidade e Cidadania: estimula a tomada de decisões com base em princípios éticos e democráticos, considerando o impacto social e ambiental das soluções propostas.

Para assegurar o desenvolvimento das competências específicas de cada área de conhecimento escolar, a cada uma delas é relacionado um conjunto de habilidades, que representa as aprendizagens essenciais que devem ser asseguradas aos alunos nos diferentes contextos escolares.

Especificamente para a área da Matemática e suas Tecnologias (Ensino Médio) e para a Matemática (Ensino Fundamental), a BNCC valoriza o desenvolvimento de habilidades como o raciocínio lógico, a capacidade de resolver problemas, de modelar situações, de comunicar ideias e de utilizar tecnologias digitais.

No contexto da robótica educacional e das metodologias ativas, a BNCC oferece um alinhamento entre ambos, de maneira natural e robusta, que pode ser observado nas dez competências descritas, e que foram potencializadas durante as práticas propostas neste estudo.

Logo, para alcançar e desenvolver essas competências e habilidades na abordagem pedagógica docente, o uso de ferramentas tecnológicas em sua prática diária, associado à aplicação de metodologias ativas, é indispensável, como veremos nas próximas seções.

PENSAMENTO COMPUTACIONAL: INTER-RELAÇÕES COM O APRENDIZADO DA MATEMÁTICA

O Pensamento Computacional (PC), conforme proposto por Wing (2006; 2011), é uma abordagem de resolução de problemas que utiliza conceitos fundamentais da ciência da computação, sendo definido por Wing (2011) como uma forma de pensar que capacita o indivíduo a formular problemas e suas soluções de maneira que um computador (seja ele um hardware ou um ser humano) possa executá-las.

Pesquisas como as de Barbosa e Maltempi (2020) apontam para os "desafios e potencialidades dos projetos de ensino e das tecnologias na formação inicial de professores", destacando como atividades "mão-na-massa" e o uso de tecnologias digitais podem articular o Pensamento Computacional com o ensino da matemática.

Logo, não se trata de ensinar a programar apenas, mas de desenvolver um conjunto de habilidades cognitivas que permitem abordar problemas complexos de forma organizada e eficiente, independentemente da área do conhecimento.



Com base em estudos como Wing (2006; 2011), Do Prado e Dantas (2024), para as atividades do curso proposto, adotamos como pilares do PC as etapas descritas a seguir:

1. Formulação do Problema: Este pilar envolve a conceituação do problema, suas necessidades, objetivos e as perguntas a serem respondidas, bem como a elaboração de planos de ação. No curso, sempre iniciamos com um "Tema Geral", que é um problema complexo e aberto, sem uma única resposta, que instiga a discussão e a investigação de subproblemas do cotidiano. Perguntas como "Podemos criar um sistema de iluminação inteligente para otimizar o consumo de energia?" , direcionada ao subproblema "Controle de Iluminação e Segurança", utilizando a placa Arduino Uno, são um desses exemplos.

2. Decomposição: É a capacidade de quebrar um problema complexo em partes menores e mais gerenciáveis. No contexto do curso, considerando, por exemplo, o tema geral "Registro de Dados Ambientais", com uso da placa Micro:bit, antes dos professores organizarem e representarem as informações dos dados de temperatura e umidade em gráficos, eles precisavam: montar o circuito, calibrar a leitura do sensor, ajustar o tempo de coleta, além de desenvolver a programação no Pictoblox, isto é, cada etapa era importante e necessária para se atingir o objetivo final.

3. Reconhecimento de Padrões: É a habilidade de identificar semelhanças, tendências ou regularidades em dados ou problemas. Particularmente para o curso, após a decomposição do tema geral (em cada encontro), nos subproblemas e placas embarcadas, era possível reconhecer que os circuitos, a programação, a compreensão e visualização das respostas eram muito similares, o que promovia uma fluidez da aprendizagem e execução das atividades em cada estação.

4. Abstração: É a capacidade de focar nas informações mais importantes, ignorando detalhes irrelevantes, e de generalizar conceitos para criar modelos. Na programação em blocos com Pictoblox, por exemplo, é exigida a "abstração de comandos", isto é, adicionar ou eliminar um bloco (ou uma sequência deles), buscando reduzir a complexidade do programa e identificando princípios gerais, faz com que o circuito lhe retorne o que realmente se pretende analisar.

5. Produção de Algoritmos: É a habilidade de desenvolver uma sequência de passos lógicos e ordenados para resolver um problema. Podem ser descritos em códigos da língua natural, matemática ou computacional. No contexto do curso, a programação em blocos exigiu a estruturação de algoritmos que envolviam operações aritméticas, comparações, lógica condicional (se, então, senão), repetições (loops), interrupções, manipulação de variáveis e detecção de sensores. Cada um para um subproblema (e placa embarcada) em específico, concretizados em blocos visuais, com respostas diretas na tela ou no circuito montado.

6. Depuração: Depurar significa buscar e corrigir erros. Esse processo também inclui testar, verificar, refinar e otimizar a solução apresentada. Durante a depuração, a organização de um processo de solução pode ser afirmado, revisado, reformulado ou abandonado. No curso, com uso do Pictoblox, a organização dos algoritmos em blocos visuais facilitou a compreensão e análise das soluções apresentadas. Consequentemente, proporciona uma melhor depuração nos códigos, quando era necessário.

Diante do exposto, no contexto da Educação Matemática, a inclusão do Pensamento Computacional na formação inicial e continuada de professores é crucial para que eles promovam em sua prática docente o desenvolvimento e fortalecimento das competências previstas na BNCC.

Vale destacar que o Pensamento Computacional é central para o "Pensamento Científico, Crítico e Criativo" (Competência 2, da BNCC), pois é uma metodologia para analisar e resolver problemas de forma sistemática. Também é intrínseco à "Cultura Digital" (Competência 5), ao promover a compreensão e o uso da tecnologia para modelar e resolver problemas.

Nas competências específicas de Matemática, o Pensamento Computacional também subsidia o desenvolvimento de habilidades de Álgebra, Lógica, Sequências e Modelagem Matemática, capacitando os alunos a criar e otimizar processos.

Na proposta do curso, como descrito, o Pensamento Computacional foi desenvolvido de forma transversal em todos os projetos. Desde a montagem de circuitos (onde a decomposição é essencial) até a programação em blocos (que exigiu o desenvolvimento de algoritmos e a abstração de comandos), os professores foram (e com certeza seus alunos serão) constantemente desafiados a pensar de maneira computacional.

Ao integrar esses conceitos com a matemática, eles não apenas resolveram problemas, mas também compreenderam a lógica por trás das soluções e como aplicá-la em diferentes contextos.

CULTURA MAKER E CONSTRUCIONISMO: ALINHAMENTO COM AS COMPETÊNCIAS DA BNCC

Para que os alunos desenvolvam as competências essenciais preconizadas pela BNCC, especialmente o Pensamento Científico, Crítico e Criativo e, a Cultura Digital, a Cultura Maker e o Construcionismo emergem como abordagens pedagógicas importantes.

A Cultura Maker, em sua essência, celebra a ideia de "fazer", "criar" e "consertar". Ela transcende o simples consumo e incentiva a proatividade, a experimentação e a prototipagem. É um convite para colocar as "mãos na massa", transformando ideias em realidade, seja através da costura, da marcenaria, da eletrônica ou da programação.



No contexto educacional, a Cultura Maker fomenta um ambiente onde os estudantes são encorajados a construir seus próprios artefatos, testar hipóteses e solucionar problemas de forma criativa, engajando-se ativamente no processo de conhecimento (como previsto na Competência 1: Conhecimento).

Complementando a Cultura Maker, o Construcionismo é uma teoria da aprendizagem desenvolvida por Seymour Papert (Papert e Harel, 1991), baseada no Construtivismo de Jean Piaget (Piaget e Greco, 1974). Enquanto Piaget postulava que o conhecimento é construído na mente do aprendiz a partir de suas interações com o mundo (construtivismo), Papert adicionou uma dimensão crucial a essa ideia: o conhecimento é construído de forma mais eficaz quando o aprendiz se engaja na construção de algo público e compartilhável.

Em suas palavras, "o construcionismo [...] é a ideia de que a construção se dá especialmente bem quando o aprendiz está engajado na construção de algo tangível, seja um castelo de areia, um motor, um programa de computador ou uma teoria" (Papert e Harel, 1991).

Para a matemática, essa conexão é essencial, contribuindo de forma significativa para o desenvolvimento do Pensamento Computacional, habilidade esta destacada por Wing (2006; 2011), visto que o pensamento computacional envolve o raciocínio lógico, a decomposição de problemas complexos, o reconhecimento de padrões e a formulação de algoritmos, elementos cruciais para a Competência 2 (Pensamento Científico, Crítico e Criativo) e para as habilidades matemáticas de modelagem e resolução de problemas.

Isto é, ao invés de simplesmente resolver problemas abstratos no papel, os alunos podem construir modelos, simulações ou jogos que exemplificam e concretizam conceitos matemáticos.

Por exemplo, a criação de um sensor de distância (utilizando Arduino Uno) para coletar dados e construir gráficos em tempo real, ou a programação de um dado eletrônico (com Micro:bit) para explorar conceitos de probabilidade, transforma a aprendizagem de uma experiência passiva para uma ativa e significativa.

Estudos como os de Pucci e Reichert (2020) e Riboldi e Reichert (2020) demonstram, por exemplo, o potencial das linguagens de programação em blocos (como o Scratch, análogo ao Pictoblox), usadas no processo de ensino de conceitos algébricos e de funções do 1º grau, respectivamente, que estimularam a investigação e a criatividade na aprendizagem de seus alunos.

Dessa forma, a Cultura Maker e o Construcionismo, ao promoverem a criação e a experimentação com tecnologias digitais (alinhando-se à Competência 5: Cultura Digital), empoderam os professores a transcender as abordagens tradicionais, convertendo a sala de aula em um laboratório de inovação.

Nesse ambiente, o erro é visto como parte do processo de aprendizagem e a criatividade é estimulada, preparando os alunos não apenas para consumir tecnologia, mas para serem criadores e transformadores em um mundo digitalizado, conforme preconizado por Valente (1993; 1999) ao discutir o papel dos computadores e softwares na educação.

APRENDIZAGEM BASEADA EM PROJETOS: MATEMÁTICA COM PROPÓSITO

A Aprendizagem Baseada em Projetos (ABP) ou, no inglês *Project-Based Learning* (PBL), é uma metodologia ativa que coloca o estudante no centro dos processos de ensino e de aprendizagem, desafiando-o a resolver problemas ou investigar questões complexas do mundo real. Essa abordagem é intrinsecamente alinhada a diversas competências da BNCC, em especial ao desenvolvimento da autonomia, do protagonismo e da capacidade de colaborar (Competências 6 e 9).

Diferente de projetos tradicionais que podem ser apenas a aplicação de um conteúdo já aprendido, na ABP, o projeto é o veículo para a aprendizagem de novos conteúdos e o desenvolvimento de habilidades. Baseado em Bender (2014), a ABP é caracterizada por alguns pilares essenciais:



1. Questão Desafiadora: Parte de um problema ou questão complexa e aberta que não possui uma única resposta e que instiga a investigação. Para a matemática, pode ser um problema do cotidiano que demande aplicação de conceitos numéricos, geométricos, estatísticos ou algébricos. Isso se alinha à capacidade de resolver problemas de maneira estruturada, usando princípios como a decomposição, o reconhecimento de padrões e a abstração, inerentes ao Pensamento Computacional (Wing, 2011) e à Competência 2 (Pensamento Científico, Crítico e Criativo) da BNCC.

2. Investigação Sustentada: Os alunos se engajam em um processo prolongado de pesquisa, experimentação e descoberta, em vez de memorizar fatos. Eles precisam buscar informações, analisar dados, aplicar conhecimentos, formular e testar hipóteses para construir a solução ou aprofundar a compreensão do problema. Este pilar garante que a aprendizagem seja ativa e baseada na curiosidade, contribuindo para o aprofundamento do Conhecimento (Competência 1) e do Pensamento Científico (Competência 2).

3. Autenticidade: Os projetos são relevantes para a vida dos alunos, com conexões com o mundo real e audiências significativas. A resolução de um problema local, por exemplo, torna a aprendizagem mais tangível e contextualizada, fomentando a Responsabilidade e Cidadania (Competência 10).

4. Voz e Escolha do Aluno: Os estudantes têm a oportunidade de expressar suas ideias e fazer escolhas sobre o produto final ou o processo de trabalho, promovendo autonomia e engajamento, elementos cruciais para o Trabalho e Projeto de Vida (Competência 6) e a Comunicação (Competência 4).

5. Reflexão: Há momentos para os alunos pensarem sobre o que estão aprendendo, como estão aprendendo e como podem melhorar, desenvolvendo o Autoconhecimento (Competência 8) e a metacognição.

6. Crítica e Revisão: O feedback de colegas e professores é parte integrante do processo, levando à melhoria contínua do trabalho, o que fortalece a Argumentação (Competência 7) e a Empatia e Cooperação (Competência 9).

7. Produto Público: O projeto culmina em um produto, apresentação ou solução que é compartilhado com uma audiência real, aumentando o senso de propósito e responsabilidade. Este produto pode ser um protótipo físico (como um sistema de monitoramento), um software, uma apresentação multimídia ou qualquer outra forma que demonstre a solução ou a compreensão adquirida, evidenciando o desenvolvimento das competências de Comunicação (Competência 4) e Cultura Digital (Competência 5).

A integração da ABP com a robótica educacional e a programação é natural, pois a construção de protótipos físicos ou digitais se alinha perfeitamente com os pilares do "Produto Público" e da "Investigação Sustentada", concretizando o "aprender fazendo" do Construcionismo. Para um projeto de matemática, a estrutura pode seguir os seguintes passos, incorporando o desenvolvimento do pensamento computacional e das competências da BNCC:

1. Definição do Problema Gerador:

- Comece com uma questão ou cenário que exija o uso de conceitos matemáticos e que possa ser resolvido ou explorado com a ajuda de uma das placas e do Pictoblox.
- Exemplo de um "problema gerador" focado em matemática: "Como podemos monitorar e otimizar o uso de água em nossa escola para reduzir o desperdício?" (implica em medição, volume, taxas, porcentagens, coleta e análise de dados). Outro exemplo diretamente dos encontros do curso: "Como podemos criar um sistema inteligente para otimizar o fluxo de veículos em cruzamentos urbanos e reduzir en-

garraamentos?" (para o tema "Controle de Tráfego" com Micro:bit). Tais problemas demandam a decomposição em partes menores e o reconhecimento de padrões, conforme o Pensamento Computacional e a Competência 2 (Pensamento Científico, Crítico e Criativo).

2. Formulação de Hipóteses e Planejamento:

- Os alunos, em grupos, discutem possíveis soluções e decidem como a matemática e os princípios do pensamento computacional podem ser usados.
- Planejam o protótipo, isto é, quais sensores ou atuadores seriam necessários? Como a programação controlaria o sistema? Quais dados seriam coletados e como seriam representados? Essa etapa envolve abstração e formulação de algoritmos, fortalecendo a Competência 2 (Pensamento Científico, Crítico e Criativo) e a Competência 6 (Trabalho e Projeto de Vida).

3. Execução (Mão na Massa):

- Programação: Utilizam o Pictoblox para codificar a lógica do protótipo. Isso pode envolver operações aritméticas, lógica condicional (se, então, senão), repetições (loops), manipulação de variáveis, etc. A concretização do algoritmo em blocos visuais facilita a compreensão e a depuração, desenvolvendo a Competência 5 (Cultura Digital).
- Montagem: Conectam os componentes eletrônicos à placa escolhida (Arduino Uno, Micro:bit ou Makey Makey). Isso envolve o reconhecimento de pinos, portas analógicas/digitais, conceitos de circuito, contribuindo para a Competência 1 (Conhecimento) e a Competência 2 (Pensamento Científico, Crítico e Criativo).
- Coleta e Análise de Dados: Se o projeto envolve sensores, os alunos coletam dados e os utilizam para fazer cálculos, criar gráficos, tabelas, e tirar conclusões matemáticas, aprofundando o Conhecimento (Competência 1) e o Pensamento Científico (Competência 2).

4. Testes e Depuração:

- Os alunos testam o protótipo, identificam erros (sejam no circuito ou no código) e os corrigem. Este é um processo iterativo que reforça o pensamento lógico, a resiliência e a habilidade de depuração, componente essencial do Pensamento Computacional e da Competência 2 (Pensamento Científico, Crítico e Criativo), além de desenvolver a Autogestão (Competência 8).

5. Apresentação e Reflexão:

- Apresentam o protótipo funcionando, explicando o problema que resolveram, os conceitos matemáticos aplicados, o processo de criação e os desafios encontrados.
- Refletem sobre a eficácia da solução e as possibilidades de aprimoramento, utilizando a Comunicação (Competência 4) e a Argumentação (Competência 7).

Ao seguir essa abordagem, a matemática deixa de ser uma disciplina isolada e se integra de forma orgânica à resolução de problemas reais, tornando o aprendizado mais profundo, significativo e memorável para os alunos, contribuindo diretamente para as competências da BNCC.

ROTAÇÃO POR ESTAÇÕES: ORGANIZANDO A SALA DE AULA PARA A EXPERIMENTAÇÃO E O DESENVOLVIMENTO DE COMPETÊNCIAS

Para viabilizar a Aprendizagem Baseada em Projetos com múltiplas tecnologias e ritmos de aprendizagem, a metodologia de Rotação por Estações (RE) surge como uma estratégia pedagógica extremamente eficaz.

Essa abordagem, ao promover o trabalho em grupo e a autonomia, fortalece diretamente a Competência 9 (Empatia e Cooperação) e a Competência 6 (Trabalho e Projeto de Vida) da BNCC. Ela permite que os alunos trabalhem em pequenos grupos, alternando entre diferentes "estações" ou centros de aprendizagem, cada um com uma atividade, tecnologia ou foco distinto.



A Rotação por Estações é um modelo flexível que pode ser adaptado a diversas realidades de sala de aula. Sua implementação requer planejamento cuidadoso para garantir que cada estação ofereça uma experiência de aprendizado valiosa e que a transição entre elas seja fluida. Baseado em Borghesan e Clement (2023) e Bacich e Moran (2018), os principais elementos para organizá-la são:

- **Definição das Estações:** Cada estação deve ter um objetivo de aprendizagem claro e materiais específicos. Pode-se ter estações focadas em diferentes tecnologias (como uma estação de Arduino Uno, outra de Micro:bit) ou em diferentes aspectos de um mesmo projeto (pesquisa, programação, montagem, prototipagem).
- **Instruções Claras:** Para que os alunos possam trabalhar de forma autônoma em cada estação, as instruções devem ser detalhadas e acessíveis (em texto, vídeo, ou com a presença de um monitor/facilitador). Isso contribui para o desenvolvimento da Comunicação (Competência 4).
- **Gerenciamento do Tempo:** Estabelecer tempos definidos para cada estação e para as transições é crucial. Ferramentas visuais (cronômetros, sinais sonoros) podem ajudar a manter o ritmo, desenvolvendo a Autogestão (Competência 8).
- **Formação dos Grupos:** Os grupos podem ser formados de maneira heterogênea (misturando diferentes níveis de habilidade) ou homogênea, dependendo dos objetivos. A formação de grupos heterogêneos é particularmente eficaz para estimular a Empatia e Cooperação (Competência 9) e o desenvolvimento da Comunicação (Competência 4).

No curso de formação de professores de Robótica Educacional com Metodologias Ativas, a Rotação por Estações foi o arcabouço prático que permitiu aos participantes explorar as diferentes placas embarcadas em um ambiente dinâmico e colaborativo, alinhando-se diretamente às competências da BNCC.

O curso foi estruturado em três encontros presenciais, e a cada encontro, a dinâmica de três estações de aprendizagem foi aplicada. A lógica foi a seguinte:

- **Encontros Temáticos e Problemas Geradores:** Cada encontro presencial teve um Tema Geral e seus Subproblemas definidos, que serviram como problemas geradores para as atividades práticas, alinhando-se à metodologia da Aprendizagem Baseada em Projetos (ABP). Isso reforça a Competência 2 (Pensamento Científico, Crítico e Criativo) e a Competência 10 (Responsabilidade e Cidadania).
 - **Encontro 1:** Foco na introdução ao Pensamento Computacional, Metodologias Ativas, Programação em Blocos e Placas Embarcadas.
 - **Encontro 2:** Tema Geral - Cidades Inteligentes. Subproblemas:
 - Controle de Iluminação e Segurança (Estação 1 - Placa Arduino Uno);
 - Controle de Tráfego (Estação 2 - Placa Micro:bit);
 - Sistemas de Alarme (Estação 3 - Placa Makey Makey).
 - **Encontro 3:** Tema Geral - Sensores e Dados para um Ambiente Sustentável. Subproblemas:
 - Temperatura, Umidade e Gases (Estação 1 - Placa Arduino Uno).
 - Registro de Dados Ambientais (Estação 2 - Placa Micro:bit).
 - Jogo da Reciclagem (Estação 3 - Placa Makey Makey).
- **Permeando pelas Estações:** Os professores, divididos em pequenos grupos, percorreram as três estações em cada encontro, dedicando tempo específico a cada uma. Isso garantiu que todos tivessem a oportunidade de interagir com cada placa (Arduino Uno, Micro:bit e Makey Makey) e desenvolver os circuitos e a programação de acordo com o subproblema da estação, promovendo a Competência 9 (Empatia e Cooperação).

- **Desenvolvimento Ativo:** Em cada estação, a tarefa dos professores era desenvolver (isto é, montar e programar) os circuitos e programas utilizando a placa embarcada respectiva, além do Pictoblox, para solucionar o subproblema proposto. Esse processo envolvia desde a montagem física do circuito e seus componentes até a escrita do código em blocos. A depuração e o teste do protótipo também faziam parte da aprendizagem. Nesta situação, fica evidenciado o desenvolvimento do Conhecimento (Competência 1) e da Cultura Digital (Competência 5).
- **Experiência Prática e Colaborativa:** A Rotação por Estações fomentou um ambiente de aprendizagem prática e colaborativa. Os professores puderam experimentar diferentes tecnologias, compartilhar conhecimentos e resolver problemas em equipe, reforçando os princípios do construcionismo. Dessa forma, ficou evidente a prática das Competências 4 (Comunicação) e 9 (Empatia e Cooperação).

Essa metodologia permitiu que, ao final do curso, os participantes tivessem uma vivência prática com as três placas, explorassem suas aplicações em diferentes contextos temáticos e compreendessem como integrar a robótica educacional e a programação em blocos ao ensino de matemática de forma ativa e envolvente, alinhada às exigências da BNCC.

ROBÓTICA EDUCACIONAL: CONSTRUINDO E PROGRAMANDO A APRENDIZAGEM

Nesse cenário de transformações no modo de pensar as práticas de ensino e aprendizagem, a Robótica Educacional (RE) tem se demonstrado uma ferramenta pedagógica enriquecedora e significativa, no que tange ao ensino integrado, haja visto que ela serve como um ponto de convergência para diversas metodologias ativas, unindo os princípios da Cultura Maker (no construir), do Construcionismo (no aprender fazendo) e do Pensamento Computacional (no programar a lógica do robô).

Outra vantagem é a abordagem de forma lúdica e desafiadora os cenários de situações-problema, que oportunizam processos de motivação, colaboração, construção e reconstrução do pensamento pessoal e computacional. Também, para programar um robô, o aluno precisa ter conhecimento integrado e amplo da tarefa a ser realizada, da estrutura mecânica do robô, dos sensores que ele possui e de seu funcionamento físico e lógico, além da clareza das tarefas que deverão ser executadas e dos conceitos a serem aplicados.



Vale destacar que a RE oferece um contexto riquíssimo para a matemática, pois a movimentação dos robôs pode empregar conceitos de geometria (ângulos, coordenadas, formas), física (velocidade, aceleração, força), lógica (condições, loops) e medição (distâncias, tempo). Integrar esses conhecimentos em um algoritmo para solucionar o problema de maneira estruturada e lógica, isto é, em um conjunto de instruções que o robô deve executar para atingir os objetivos da sua construção, torna os conceitos matemáticos mais concretos e acessíveis.

A RE potencializa várias competências gerais da BNCC, destacando-se a "Pensamento Científico, Crítico e Criativo" (Competência 2), ao fomentar a resolução de problemas e a investigação; a "Cultura Digital" (Competência 5), pela manipulação de hardware e software e o desenvolvimento de soluções tecnológicas; o "Trabalho e Projeto de Vida" (Competência 6), pela execução de projetos complexos; a "Comunicação" (Competência 4) e a "Cooperação e Colaboração" (Competência 9), no trabalho em equipe e na apresentação dos resultados. Especificamente na Matemática, ela se conecta com habilidades de Geometria, Grandezas e Medidas, Álgebra (sequências e lógica) e Estatística (análise de dados de sensores e movimentos).

Desse modo, o papel do docente, como mediador desse processo de construção, é essencial para que os objetivos propostos sejam traçados e alcançados. Podemos destacar algumas expertises a serem promovidas pelos docentes, quando eles se utilizam da RE em projetos interdisciplinares de ensino-aprendizagem:

- tornam a aprendizagem em um momento motivador, explanam de forma mais acessível os princípios da ciência e tecnologia;
- permitem experiências por meio de equipamentos físicos que simulam o mundo real;
- promovem o desenvolvimento do raciocínio e da lógica na construção de algoritmos e programas para controle de mecanismos;

- propõem ao aluno a gestão de seus sentimentos e expressão, quando ele aprende a comunicar e verbalizar seus conhecimentos e experiências, desenvolvendo assim a capacidade de argumentar e gerar seu próprio senso crítico;
- estimulam invenções por meio do design mecânico;
- além de promoverem a interdisciplinaridade por meio da integração de conceitos das mais diversas áreas envolvidas.

Logo, ao envolver os professores (e posteriormente, os alunos), em desafios reais e permitindo que eles construíssem e programassem suas próprias soluções, este material, fruto do curso de formação docente, buscou não somente ensinar matemática, mas orientar professores a desenvolverem sua criatividade, criticidade e expertise para os desafios do mundo contemporâneo.

Em resumo, a combinação estratégica da Cultura Maker, Construcionismo, Aprendizagem Baseada em Projetos, Rotação por Estações, Pensamento Computacional e Robótica Educacional oferece um ambiente de aprendizagem dinâmico e eficaz.



CAPÍTULO 2

CONHECENDO AS FERRAMENTAS



PICTOBLOX: A PORTA DE ENTRADA PARA A PROGRAMAÇÃO



O Pictoblox é um software de programação baseado em blocos, construído sobre a plataforma Scratch. Ele serve como uma ponte intuitiva entre a programação visual e o controle de hardware, permitindo que usuários de todas as idades, especialmente iniciantes, criem projetos interativos sem a necessidade de aprender linguagens de código complexas. Sua interface gráfica, onde comandos são representados por blocos coloridos que se encaixam como peças de Lego, desmistifica o processo de codificação.

O Pictoblox foi desenvolvido pela Stempedia, uma empresa indiana com foco em educação STEM (Ciência, Tecnologia, Engenharia e Matemática). A motivação por trás de sua criação foi tornar o aprendizado de conceitos avançados, como Inteligência Artificial, robótica e Internet das Coisas (IoT), acessível a crianças e adolescentes. Ao basear-se no Scratch, já conhecido por sua interface intuitiva, o Pictoblox buscou adicionar camadas de funcionalidade que o conectam a hardware real e a tecnologias emergentes de forma simplificada.

COMO FUNCIONA

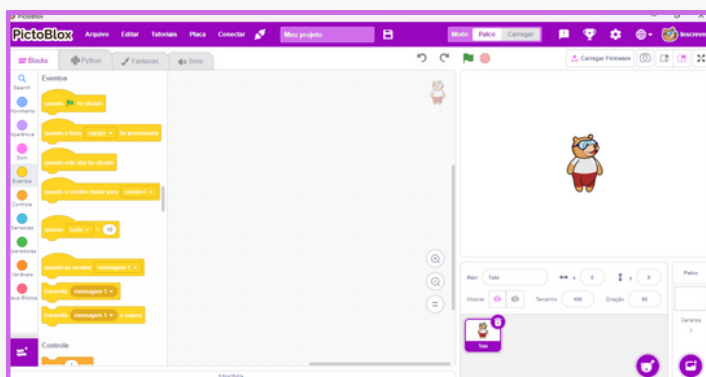
A interface do Pictoblox é amigável e dividida em áreas principais:

01 PALCO:
Onde os projetos ganham vida. É aqui que os personagens (sprites) se movem, interagem e onde os resultados das programações são visualizados.

02 SPRITES:
Personagens ou objetos que podem ser programados. O Pictoblox oferece uma biblioteca vasta e permite importar novos.

03 ÁREA DE SCRIPT:
O espaço em branco onde os blocos são arrastados e encaixados para formar os programas.

04 ÁREA DE BLOCOS:
Agrupa os blocos de programação por categoria (Movimento, Aparência, Som, Eventos, Controle, Sensores, Operadores, Variáveis e Meus Blocos). Cada categoria tem uma função específica, como controlar o movimento de um sprite ou realizar operações lógicas.



05 CONEXÃO DE DISPOSITIVOS:

Uma das grandes vantagens do Pictoblox é sua capacidade de se conectar facilmente a diversas placas embarcadas, incluindo Arduino Uno, Micro:bit e Makey Makey. Isso permite que a programação visual controle elementos físicos, como LEDs, motores e sensores.



Ainda, o Pictoblox se destaca pela sua integração com a Inteligência Artificial (IA). Ele incorpora blocos que permitem a utilização de conceitos de aprendizado de máquina, reconhecimento de voz e visão computacional em projetos simples. Isso abre um leque de possibilidades para a criação de interações mais avançadas e para explorar a IA de forma prática e acessível, mesmo em projetos matemáticos (por exemplo, um projeto que reconhece gestos para resolver equações ou um sistema de visão para contar objetos).

POR QUE USAR PROGRAMAÇÃO EM BLOCOS COM INICIANTES

A programação em blocos é a porta de entrada ideal para o universo da codificação, especialmente para educadores e alunos iniciantes. Suas principais vantagens incluem:

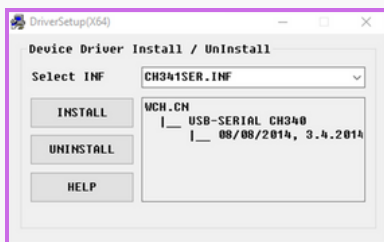
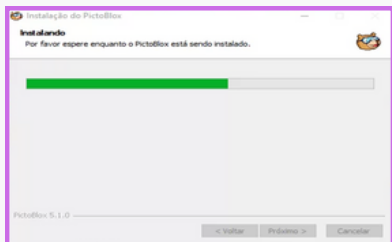
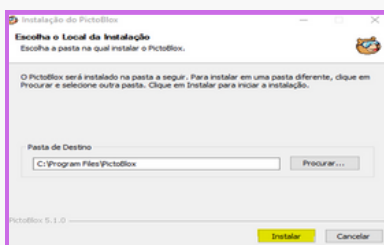
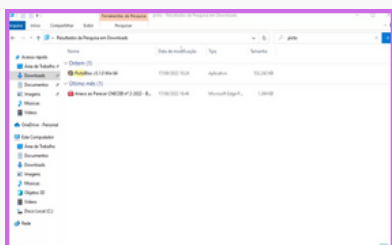
- **Redução da Curva de Aprendizagem:** A natureza visual dos blocos elimina a necessidade de memorizar sintaxe complexa, permitindo que o foco seja na lógica do programa.
- **Menos Erros de Sintaxe:** Como os blocos se encaixam apenas de maneiras válidas, a maioria dos erros de sintaxe é prevenida, o que reduz a frustração e acelera o processo de aprendizagem.
- **Foco no Pensamento Computacional:** Ao invés de se preocupar com os detalhes da linguagem, os usuários podem se concentrar nos conceitos essenciais do pensamento computacional: decomposição de problemas, reconhecimento de padrões, abstração e algoritmos.
- **Feedback Imediato:** As mudanças no código são imediatamente visíveis no palco ou no comportamento do hardware conectado, proporcionando um feedback instantâneo que facilita a compreensão e a depuração.

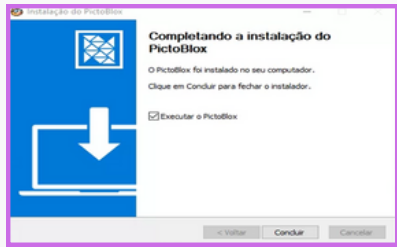
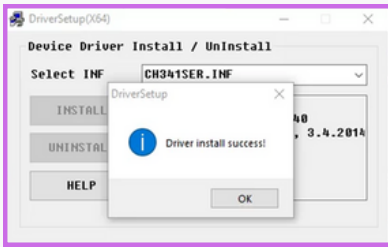
Para o ensino da matemática, o Pictoblox permite que conceitos abstratos sejam visualizados e manipulados. Gráficos, funções, probabilidade, geometria e até mesmo noções de estatística podem ser explorados de forma interativa, tornando a matemática mais tangível e divertida.

COMO OBTER E COMUNIDADE

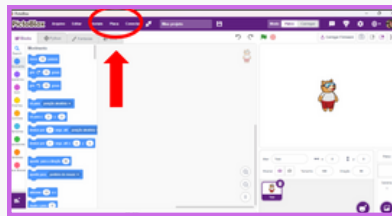
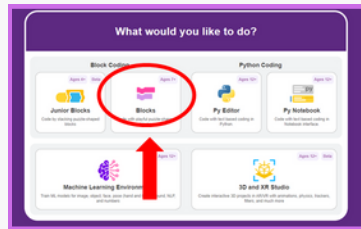
- **Download:** O Pictoblox é um software gratuito e pode ser baixado diretamente do site oficial da Stempedia. Uma pesquisa rápida por "Pictoblox download" levará à página de instalação para diferentes sistemas operacionais (Windows, macOS, Linux, Android e iOS).
- **Instalando o Pictoblox no Windows:** separamos três endereços confiáveis que o leitor poderá utilizar para baixar o Pictoblox para o sistema operacional Windows:
 - <https://thestempedia.com/product/pictoblox/download-pictoblox/>
 - <https://www.blogdarobotica.com/2021/07/23/guia-de-instalacao-do-pictoblox/>
 - <http://www.computacaofisica.com.br/>

Observe o roteiro a seguir:

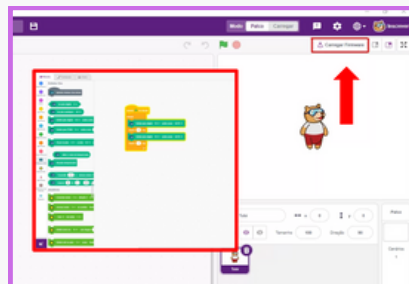
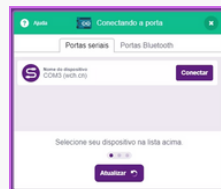




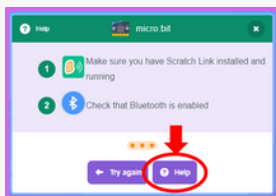
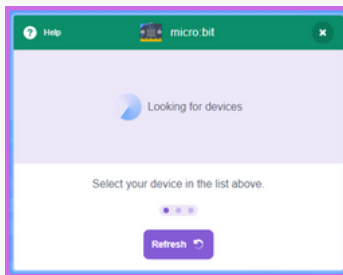
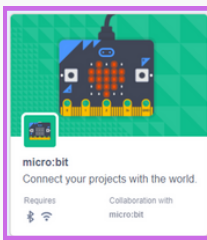
- **IDEs (Ambientes de Desenvolvimento):** A própria aplicação Pictoblox funciona como a IDE principal. Veja a seguir a habilitação de cada placa no Pictoblox:



- **Arduino Uno:**

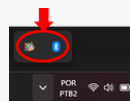
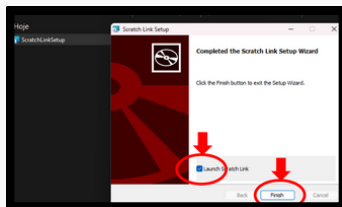
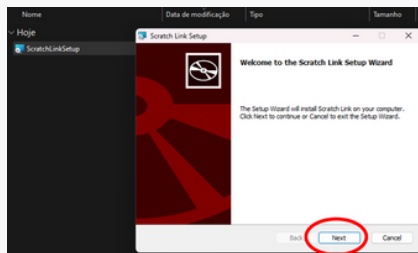
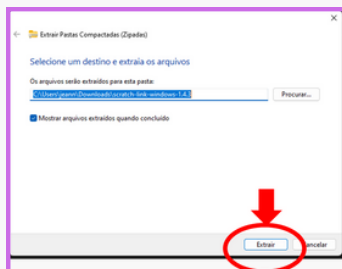
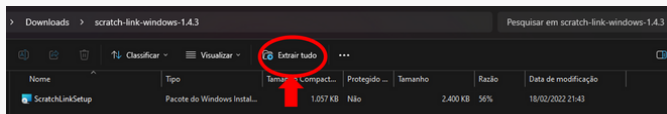


o Micro:bit:



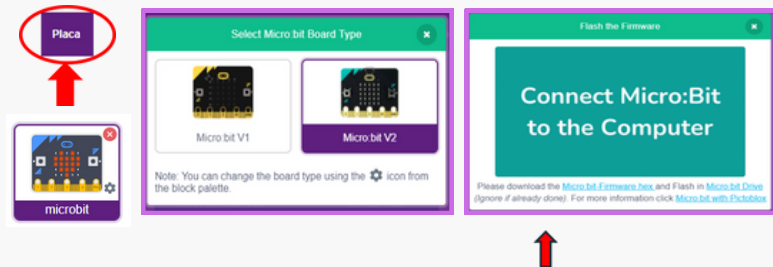
<https://drive.usercontent.google.com/download?id=1AFaLkdS1jwvQINxzV2EdcvdpcC488Nf&export=download&authuser=0>

| Nome | Data de modificação | Tipo | Tamanho |
|----------------------------|---------------------|------------------|----------|
| Hoje | | | |
| scratch-link-windows-1.4.3 | 04/09/2024 10:33 | Pasta compactada | 1.057 KB |

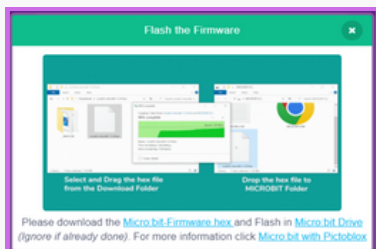
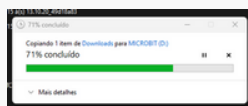
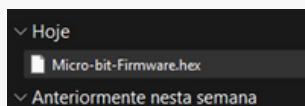


O Pictoblox Link deve se automaticamente ativado (sempre que for usar a BBC Micro:bit)

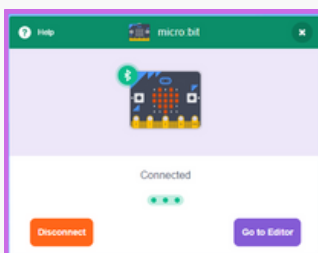
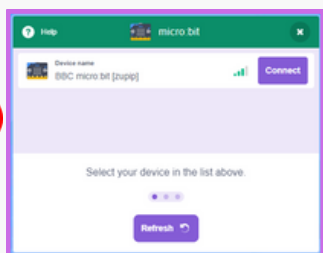
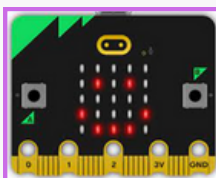
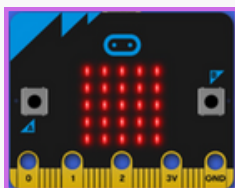
- Se NÃO:
 1. Clique no menu iniciar e procure por "Pictoblox Link"
 2. Clique no aplicativo para abri-lo



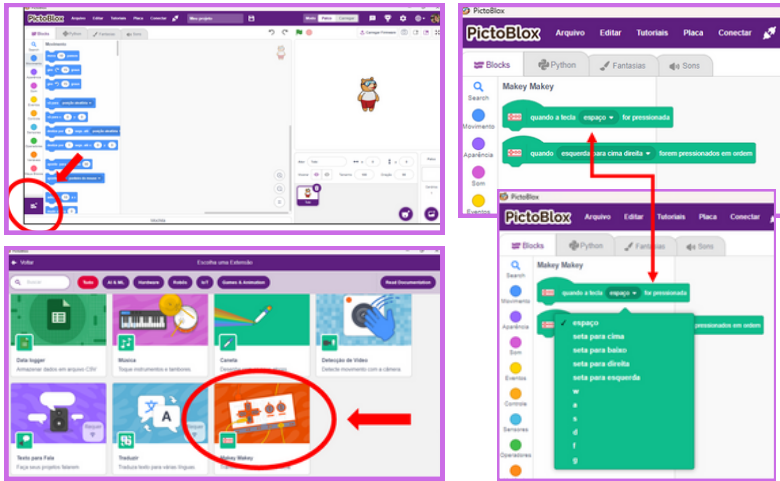
Baixar o Micro-bit-Firmware e arrastá-lo para a micro:bit.
 Essa ação deve ser feita uma única vez. Guarde o firmware
 no computador e use sempre com a micro:bit e o Pictoblox.



Observar a matriz de LEDs da micro:bit que
 indicará a ação para o pareamento: "tilt to fill
 screen (incline para preencher a tela)".



○ Makey Makey:

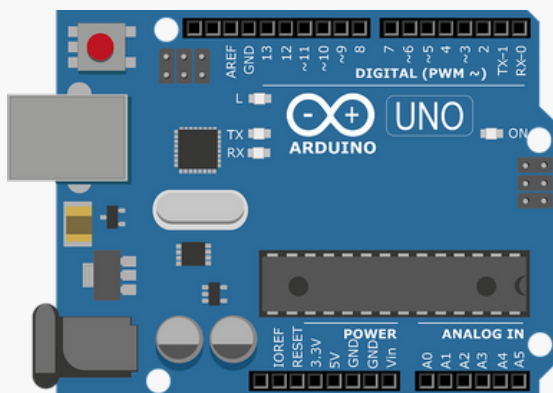


- **Comunidades e Recursos:** A Stempedia oferece documentação extensa, tutoriais e uma comunidade online para suporte. Além disso, por ser baseado no Scratch, muitas estratégias e projetos desenvolvidos na vasta comunidade Scratch podem ser adaptados para o Pictoblox, ampliando as possibilidades de aprendizado e troca.

• (Alguns) Links para exploração:

- <https://ai.thestempedia.com/docs/pictoblox/arduino-uno-with-pictoblox>
- <http://www.computacaofisica.com.br/>
- <https://www.blogdarobotica.com/category/arduino/pictoblox/>
- <https://www.instructables.com/How-to-Interface-Arduino-Board-to-PictoBlox-Scratch/>
- https://www.youtube.com/watch?v=8O_tQN-IR4g
- <https://dalousaarobotica.com.br/blog/blocos-para-arduino/>
- <https://www.youtube.com/watch?v=36hpVDMHeYU>
- https://www.youtube.com/channel/UCCGk83PAQ5aGR7IVID_cBaw
- https://www.youtube.com/channel/UCKHhA5hN2UohhFDfNXB_cvQ

ARDUINO: VERSATILIDADE E POTENCIAL



O Arduino Uno é uma placa de prototipagem eletrônica de código aberto que combina hardware (placas de microcontroladores) e software (um ambiente de desenvolvimento integrado - IDE). Ele é amplamente utilizado por iniciantes e profissionais para criar projetos que interagem com o mundo físico através de sensores e atuadores, devido sua robustez e facilidade de uso. Sua versatilidade o torna uma ferramenta poderosa para o ensino de ciências, tecnologia, engenharia, artes e matemática. No contexto deste material, focaremos nas placas Arduino Uno.

O projeto Arduino foi criado em 2005 por Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis no Interaction Design Institute Ivrea (IDII), na Itália. A principal motivação era fornecer uma ferramenta de baixo custo e fácil de usar para estudantes de design e artes que não tinham conhecimento em eletrônica e programação, permitindo-lhes criar protótipos interativos rapidamente. O nome "Arduino" vem de um bar local onde alguns dos fundadores se encontravam. A filosofia de código aberto foi central para o seu rápido crescimento e adoção global.

PRINCIPAIS COMPONENTES

As placas Arduino são basicamente pequenos computadores que podem ser programados para ler informações de sensores e controlar atuadores.

01 MICROCONTROLADOR:

O "cérebro" da placa. É responsável por executar o código.

02 CONEXÃO USB:

Utilizada para carregar o programa para a placa e para a comunicação serial com o computador.

03 PORTAS DIGITAIS:

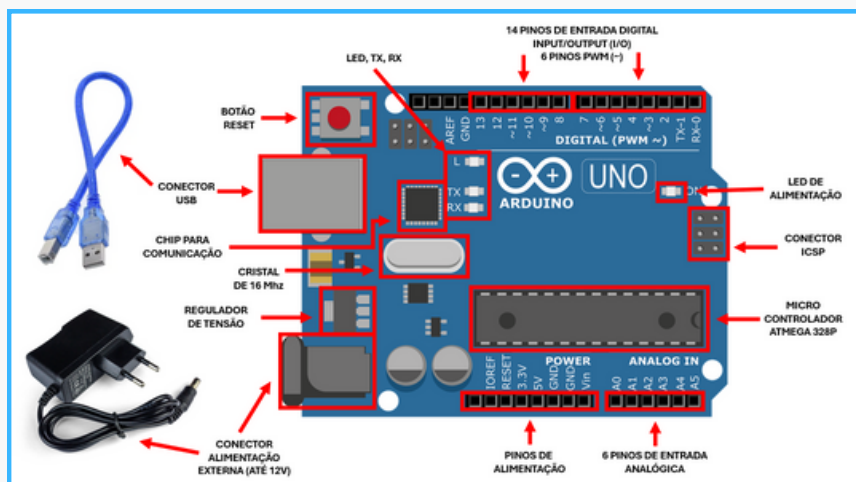
Pinagem que pode ser configurada como entrada ou saída. Usadas para ligar (ou desligar) componentes (LEDs, motores), ou ler sinais digitais (botões). O Arduino Uno tem 14 pinos digitais.

04 PORTAS ANALÓGICAS:

Usadas para ler sinais analógicos, como a variação de luz de um sensor de luminosidade ou a leitura de um potenciômetro. O Arduino Uno tem 6 pinos analógicos.

05 PINOS DE ALIMENTAÇÃO:

Fornecem energia para os componentes externos. GND (ground ou terra) é o polo negativo. 5V e 3.3V fornecem voltagens positivas.



SENSORES E ATUADORES BÁSICOS

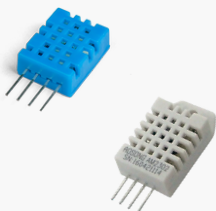
A interação do Arduino com o ambiente se dá através de sensores (que "sentem" o mundo) e atuadores (que "agem" no mundo), interligando componentes de forma simples, por meio de uma placa de ensaio (protoboard), sem a necessidade de solda.

Ao integrar o Arduino a projetos de matemática, os professores podem tornar conceitos abstratos em experiências tangíveis, permitindo que os alunos visualizem dados, simulem cenários e apliquem a matemática para resolver problemas práticos, como visto nos desafios de "Controle de Iluminação e Segurança" e "Temperatura, Humidade e Gases" do curso oferecido.

- **Sensores:**

- **Sensor de Distância Ultrassônico (HC-SR04):**

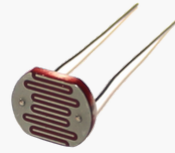
Emite ondas sonoras e mede o tempo que levam para retornar, calculando a distância de um objeto. Aplicação matemática: coleta de dados para construção de gráficos de posição-tempo, análise de funções de distância.



- **Sensor de Temperatura e Umidade (DHT11 e DHT22):**

Mede a temperatura e a umidade do ar. Aplicação matemática: coleta de dados para estatística, construção de séries temporais, análise de variações térmicas.

- **Sensor de Luz (LDR - Resistor Dependente de Luz):** Varia sua resistência com a intensidade da luz. Aplicação matemática: medir intensidade de luz para experimentos de proporção, ou controlar a luminosidade de um LED com base em um cálculo.



- **Potenciômetro:** Resistor variável que permite controlar uma voltagem, ideal para entrada de dados analógicos. Aplicação matemática: explorar funções lineares (volt-resistência), controle de variáveis em equações.



- **Atuadores:**

- **LEDs (Light Emitting Diodes):** Diodos que emitem luz. Podem ser ligados/desligados ou ter sua intensidade variada (PWM). Aplicação matemática: representar estados binários (verdadeiro/falso), simular semáforos em problemas de tráfego, explorar conceitos de contagem e padrões.



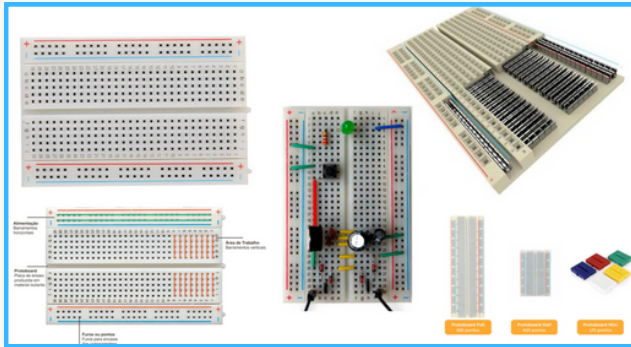
- **Buzzer (Alto-falante):** Produz sons. Aplicação matemática: criar sequências sonoras baseadas em padrões numéricos, explorar frequências e ritmo.



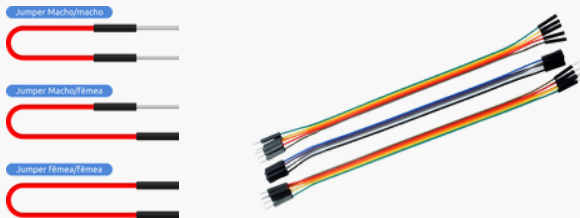
- **Servomotor:** Motor que pode ser girado para posições específicas. Aplicação matemática: ângulos, geometria, construção de relógios ou braços robóticos para representar vetores.



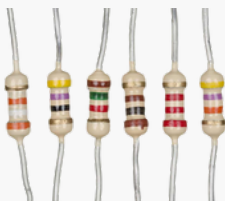
- **Protoboard:** é uma placa de ensaio que serve como um protótipo de um aparelho eletrônico, com uma matriz de contatos que possibilita construir circuitos de teste sem que haja necessidade de solda e, assim, garantindo segurança e agilidade em diferentes atividades.



- **Jumper:** é um pedaço de fio ou um componente que conecta dois pontos em um circuito, permitindo que a corrente elétrica flua entre eles. São frequentemente usados para configurar placas de circuito ou para fazer conexões temporárias entre componentes.



- **Resistor:** a principal função de um resistor é oferecer resistência à passagem da corrente elétrica em um circuito.



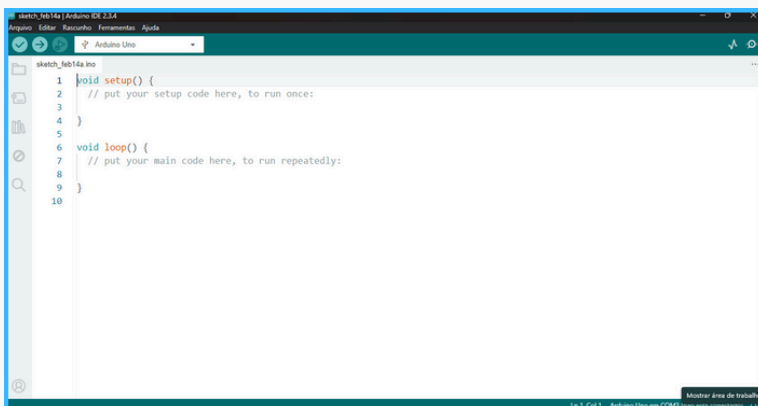
Código de Cores

A extremidade com mais faixas deve apontar para a esquerda

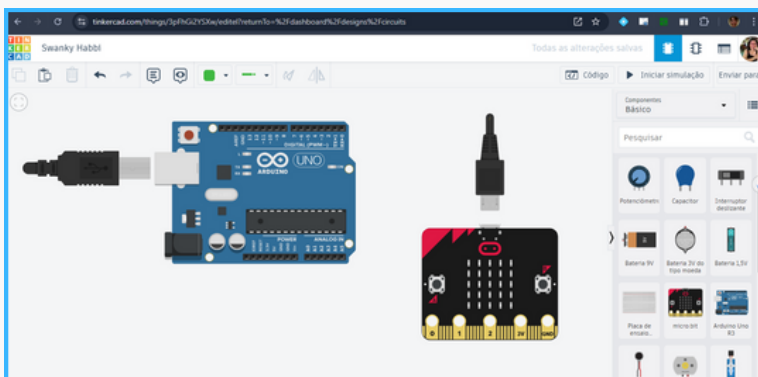
| Cor | 1ª Faixa | 2ª Faixa | 3ª Faixa | Multiplicador | Tolerância |
|----------|----------|----------|----------|---------------|------------|
| Preto | 0 | 0 | 0 | x 1 Ω | |
| Marrrom | 1 | 1 | 1 | x 10 Ω | ±1% |
| Vermelho | 2 | 2 | 2 | x 100 Ω | ±2% |
| Laranja | 3 | 3 | 3 | x 1K Ω | |
| Amarelo | 4 | 4 | 4 | x 10K Ω | |
| Verde | 5 | 5 | 5 | x 100K Ω | ±0.5% |
| Azul | 6 | 6 | 6 | x 1M Ω | ±0.25% |
| Violeta | 7 | 7 | 7 | x 10M Ω | ±0.1% |
| Grises | 8 | 8 | 8 | | ±0.05% |
| Branco | 9 | 9 | 9 | | |
| Dourado | | | | x 1 Ω | ±5% |
| Prateado | | | | x 0.1 Ω | ±10% |

COMO OBTER E COMUNIDADE

- **Download da IDE:** O ambiente de desenvolvimento (IDE) do Arduino é gratuito e pode ser baixado em
 - <https://www.arduino.cc/en/software>.
- Existe uma versão instalável para desktop e um Editor Web para programar diretamente no navegador.



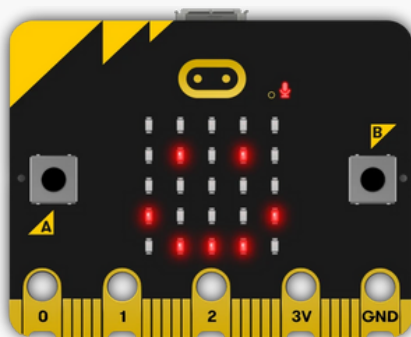
- **Tinkercad** (<https://www.tinkercad.com/>): é uma plataforma online, gratuita, de modelagem 3D, design e simulação de circuitos eletrônicos, especialmente voltada para iniciantes e educação. Ele permite que usuários criem modelos 3D, simulem circuitos eletrônicos e até programem para robótica, com placas virtuais como o Arduino e a Micro:bit.



- **Comunidades e Recursos:** A comunidade Arduino é uma das maiores e mais ativas do mundo maker.
 - **Fórum Oficial do Arduino:** <https://forum.arduino.cc/> (ótimo para dúvidas e projetos).
 - **Documentação e Tutoriais:** O site oficial do Arduino possui uma vasta biblioteca de exemplos e guias.
 - **YouTube e Blogs:** Inúmeros canais e blogs dedicados a projetos com Arduino.
 - **Plataformas Educacionais:** Muitas plataformas (como Tinkercad, para simulação) suportam o Arduino, facilitando o aprendizado.

- **(Alguns) Links para exploração:**
 - <https://www.arduino.cc/>
 - <https://www.blogdarobotica.com/2020/09/17/conhecendo-a-placa-arduino-uno/>
 - <https://docs.arduino.cc/learn/starting-guide/getting-started-arduino/>
 - <https://www.makerhero.com/blog/category/arduino/>
 - <https://blog.eletrogate.com/arduino-primeiros-passos/>
 - <https://www.makerhero.com/blog/arduino-ide/>
 - <https://www.makerhero.com/blog/aprenda-a-piscar-um-led-com-arduino/>
 - <https://www.makerhero.com/blog/como-criar-um-semaforo-com-arduino/>
 - <https://www.robocore.net/tutoriais/arduino/>
 - <https://www.makerhero.com/blog/arduino-web-editor-ide-online/>
 - <https://www.blogdarobotica.com/category/arduino/>
 - <https://blog.eletrogate.com/braco-robotico-mdf-bluetooth/>
 - <https://www.youtube.com/watch?v=sv9dDtYnE1g>

MICRO:BIT: SIMPLES E PRÁTICA



A Micro:bit é uma pequena placa de computador programável, que foi desenvolvida pela British Broadcasting Corporation (BBC) e lançada em 2016 no Reino Unido. Milhões de placas foram distribuídas gratuitamente para crianças em idade escolar, com o objetivo de combater a escassez de habilidades digitais e promover a criatividade com a tecnologia.

A motivação principal era inspirar uma nova geração de crianças a se tornarem "inovadores digitais" e a desenvolverem habilidades em ciência, tecnologia, engenharia e matemática. A iniciativa foi um sucesso e a Micro:bit se tornou um fenômeno global na educação. Seu design compacto e recursos integrados a tornaram uma ferramenta excelente para introduzir conceitos de programação e eletrônica, especialmente para crianças e jovens.

Diferentemente do Arduino, que requer mais componentes externos, a Micro:bit já vem com diversos sensores e atuadores embutidos, facilitando o início de projetos rápidos e visualmente impactantes. Os desafios do curso sobre "Controle de Tráfego" e "Registro de Dados Ambientais" demonstram sua aplicabilidade.

RECURSOS INTEGRADOS

A Micro:bit é uma placa "tudo em um", incluindo:

01 MATRIZ DE LEDS 5X5:

Grade de 25 pequenos LEDs que podem ser programados para exibir letras, números, ícones ou animações simples.

02 PINOS DE CONEXÃO (I/O):

Pinos que permitem a conexão de componentes externos, como sensores e atuadores adicionais.

03 BOTÕES PROGRAMÁVEIS:

Os botões A e B são entradas digitais que podem ser usadas para interagir com o programa.

04 BÚSSOLA (MAGNETÔMETRO):

Sensor que detecta campos magnéticos e pode ser usado como uma bússola digital.

05 SENSOR DE TEMPERATURA:

Sensor interno que mede a temperatura ambiente da placa.

06 SENSOR DE LUMINOSIDADE:

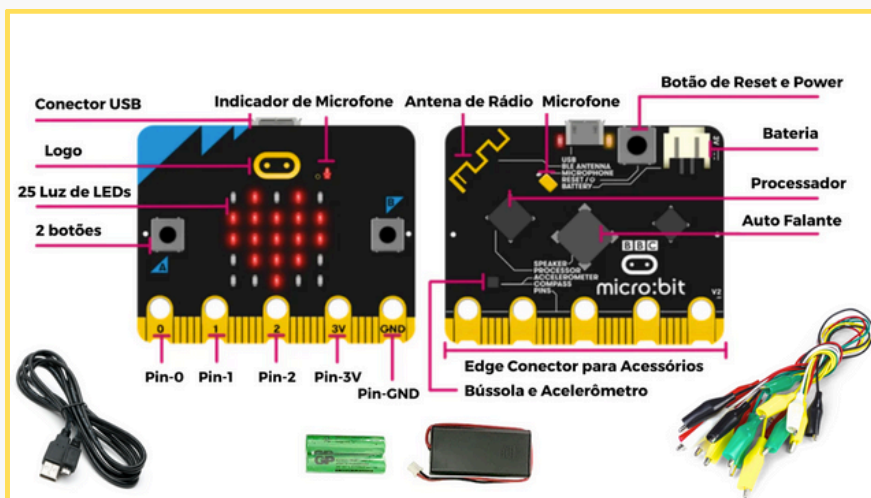
Sensor interno que mede a luminosidade do ambiente externo ao redor da placa.

07 BLUETOOTH:

Permite a comunicação sem fio com outros dispositivos.

08 ACELERÔMETRO:

Sensor que detecta movimento, inclinação e vibração.



IDEAL PARA PROJETOS RÁPIDOS E VISUALIZAÇÃO DE DADOS

A simplicidade e os recursos integrados da Micro:bit a tornam perfeita para:

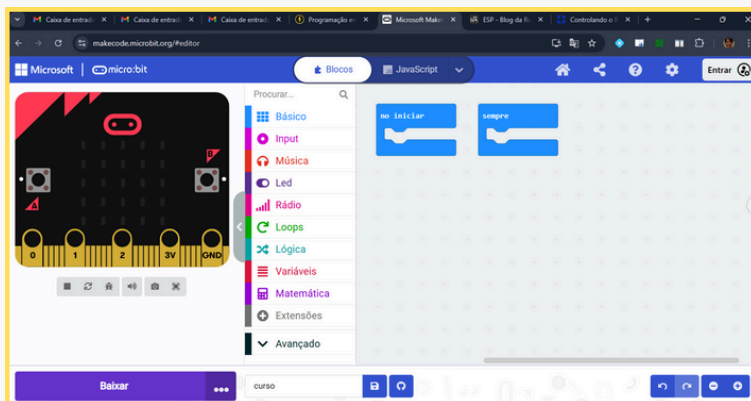
- **Projetos de Início Rápido:** A facilidade de uso e a ausência de fios complexos permitem que os alunos comecem a programar e ver resultados em minutos.
- **Visualização de Dados e Feedback Instantâneo:** A matriz de LEDs é excelente para exibir resultados numéricos ou gráficos simples, oferecendo feedback visual imediato para conceitos matemáticos.
- **Exemplos de Projetos para a Matemática:**
 - Dado Eletrônico: Programar a Micro:bit para exibir um número aleatório de 1 a 6 na matriz de LEDs ao ser agitada (usando o acelerômetro e a função de aleatoriedade). Conceitos: probabilidade, números aleatórios.
 - Contador de Passos/Ciclos: Usar o acelerômetro para contar movimentos e exibir o total. Conceitos: contagem, estatística básica.
 - Termômetro Digital: Exibir a temperatura ambiente na matriz de LEDs. Conceitos: medidas, variação de dados, gráficos de temperatura.
 - Calculadora Simples: Programar os botões para realizar operações aritméticas básicas. Conceitos: operações fundamentais.
 - Matriz de LEDs 5x5: Aplicação matemática na visualização de padrões numéricos, coordenadas cartesianas e representação gráfica de dados binários.
 - Botões Programáveis (A e B): Aplicação matemática na contagem, seleção de opções em jogos e entradas para equações.
 - Acelerômetro: Aplicado na medição da aceleração para experimentos de física, detecção de orientação (ângulos) e criação de jogos que respondem ao movimento.
 - Bússola (Magnetômetro): Auxilia nas noções de orientação espacial e coordenadas polares.

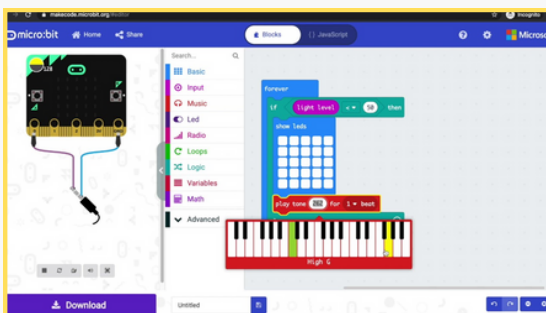
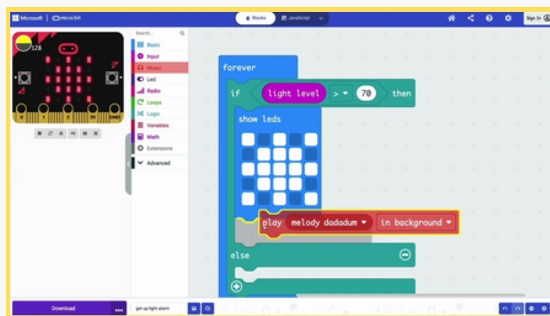
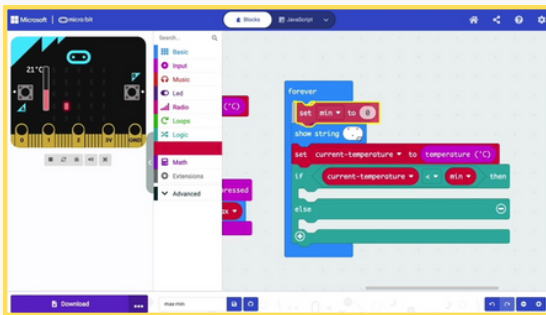
- Sensor de Temperatura: Utilizado na coleta de dados para gráficos e estatística.
- Bluetooth: Utilizado na transmissão de dados e controle remoto de outros projetos.

A Micro:bit se apresenta como uma ferramenta poderosa para tornar a matemática mais interativa e contextualizada, permitindo que os alunos explorem conceitos matemáticos de forma lúdica e prática.

COMO OBTER E COMUNIDADE

- **Download/Ambientes Online:** A Micro:bit é programada principalmente através de ambientes de desenvolvimento baseados em navegador, o que elimina a necessidade de downloads e instalações complexas. Sua principal IDE é o Microsoft MakeCode Editor, além do MicroPython Editor.
 - <https://makecode.microbit.org/> (ambiente principal, com programação em blocos e JavaScript).
 - <https://python.microbit.org/> (para programação em código Python).
 - O Pictoblox também permite programar a Micro:bit com blocos.



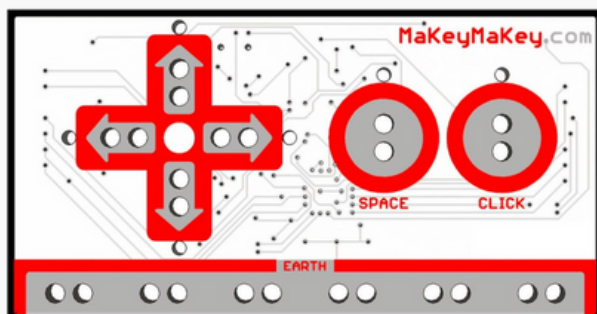


- **Comunidades e Recursos:** A Micro:bit Educational Foundation (criada após o sucesso do projeto da BBC) lidera o desenvolvimento e a disseminação da placa, com uma vasta rede de educadores e recursos.
 - Site Oficial (Recursos Educacionais): <https://microbit.org/> (tutoriais, atividades, guias para educadores).
 - Fóruns e Comunidades de Educadores: Existem muitos grupos de educadores em plataformas como Facebook, Twitter e comunidades online dedicadas ao uso da Micro:bit em sala de aula.

- **(Alguns) Links para exploração:**

- <https://microbit.org/>
- <https://microbit.org/get-started/features/overview/>
- <https://www.makerhero.com/guia/microbit/>
- <https://www.robocore.net/tutoriais/bbc-microbit/>
- <https://www.blogdarobotica.com/category/microbit/>
- <https://www.robocore.net/tutoriais/programacao-em-blocos-para-microbit>
- <https://microbit.org/projects/make-it-code-it/>
- <https://www.youtube.com/watch?v=TV8tI05Kdmw&t=13s>
- <https://www.tinkercad.com/>
- https://www.youtube.com/watch?v=VPtWn-_LbzI
- <https://www.makerhero.com/blog/simulando-a-plataforma-microbit/>
- <https://www.youtube.com/watch?v=CrHj4OQ6Sw&t=550s>
- <https://www.makerhero.com/guia/tinkercad/>
- <https://python.microbit.org/v/3>

MAKEY MAKEY: TRANSFORMANDO O COTIDIANO EM INTERFACE



A Makey Makey é uma placa de prototipação simples, que transforma objetos do cotidiano em teclas de teclado ou botões de mouse. Sua premissa é genial: se um objeto conduz eletricidade, ele pode ser usado como um controle para interagir com um computador. É uma ferramenta excepcional para despertar a criatividade, especialmente em contextos que envolvem arte, música e, de forma surpreendente, matemática. No curso, vimos sua aplicação em projetos como "Sistemas de Alarme" e "Jogo da Reciclagem".

O Makey Makey foi desenvolvida por Jay Silver e Eric Rosenbaum, enquanto eram estudantes no MIT Media Lab. Lançado em 2012 através de uma campanha de *crowdfunding*, o projeto nasceu da ideia de que qualquer coisa pode ser uma entrada para o computador, bastando que conduza eletricidade. A motivação era democratizar a invenção e a interação digital, tornando a criação tecnológica mais divertida, acessível e menos intimidante para pessoas de todas as idades, independentemente de sua experiência prévia em eletrônica ou programação. Eles queriam que as pessoas pudessem "brincar com o mundo" de uma forma nova e expressiva.

COMO FUNCIONA (CIRCUITO ABERTO/FECHADO)

O Makey Makey funciona com base no princípio de circuito elétrico aberto e fechado. Ele possui entradas para os "elementos condutores" (os objetos que você quer transformar em botões) e uma saída de "Terra" (Ground).

01 CONEXÃO USB:

A Makey Makey é conectada ao computador via USB, sendo reconhecida como um teclado e mouse comuns.

02 CONECTANDO OS OBJETOS:

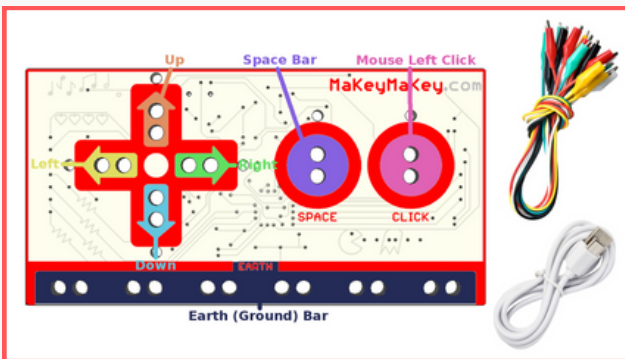
Você conecta um cabo jacaré (tipo "clip") da placa Makey Makey a um objeto condutor (ex: banana, massinha, grafite). Este objeto se torna uma das entradas, como uma tecla "espaço" ou uma "seta".

03 FECHANDO O CIRCUITO:

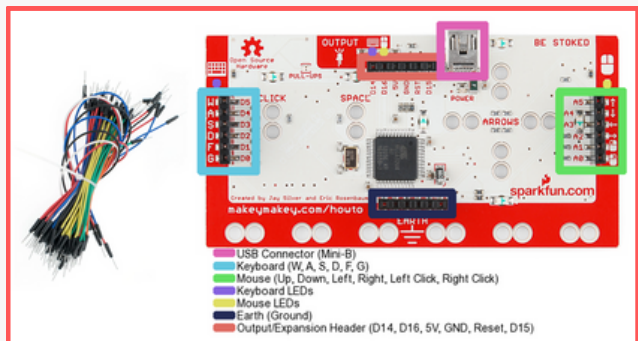
Para que o "botão" funcione, o usuário deve segurar o cabo de "terra" (ground) com uma mão (ou parte do corpo) e tocar o objeto condutor com a outra. Ao tocar o objeto, o corpo do usuário fecha o circuito elétrico entre a Makey Makey e o objeto condutor.

04 ENVIO DE SINAL:

A Makey Makey detecta o fechamento do circuito e envia o sinal correspondente para o computador. O softwares como Pictoblox, Scratch, ou qualquer outro programa que responda a toques de teclado/mouse, reage a esse sinal.



OBS: É importante destacar que a corrente elétrica utilizada é baixíssima e completamente segura para o toque humano.



IDEIAS DE APLICAÇÕES PRÁTICAS

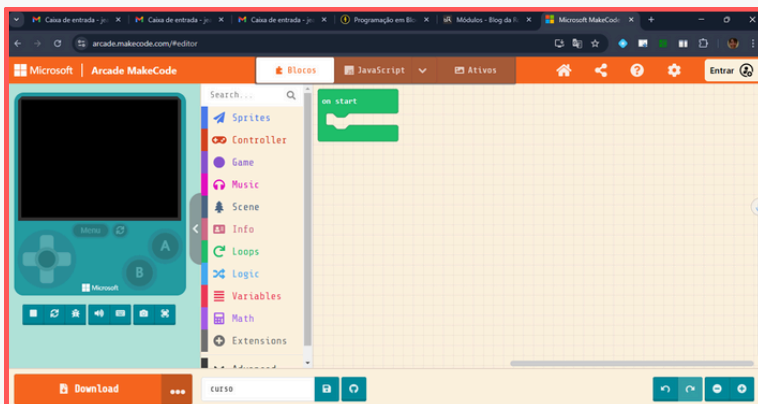
A simplicidade e a inovação do Makey Makey abrem um universo de possibilidades criativas e matemáticas:

- **Calculadoras Interativas com Massinha:** Crie botões de massinha condutora para os números e operações. Ao tocar a massinha, o Pictoblox (ou Scratch) recebe o sinal e realiza o cálculo, exibindo o resultado. Conceitos: operações aritméticas, lógica, representação de números.
- **Jogos de Geometria com Desenhos a Grafite:** Desenhe formas geométricas em papel com grafite (que é condutor). Ao tocar as linhas, o Makey Makey pode disparar sons, animações ou perguntas no Pictoblox sobre as propriedades das formas. Conceitos: geometria, propriedades de figuras, coordenadas.
- **Gráficos Interativos:** Crie um gráfico com materiais condutores onde tocar diferentes pontos aciona eventos ou exibe valores relacionados a funções ou dados.
- **Controladores para Jogos de Matemática:** Construa joysticks ou botões personalizados para jogos desenvolvidos no Pictoblox que trabalham com tabuada, raciocínio lógico ou desafios de frações.
- **Linha Numérica Tátil:** Crie uma linha numérica no chão com materiais condutores. Conecte cada número a uma entrada do Makey Makey. Os alunos podem "pular" para um número específico, e o Pictoblox pode exibir informações ou fazer perguntas relacionadas àquele número (ex: "é par/ímpar?", "é múltiplo de X?").

A Makey Makey é uma ferramenta poderosa para tornar o aprendizado da matemática uma experiência sensorial e colaborativa, permitindo que os alunos criem interfaces lúdicas e explorem conceitos de forma ativa e divertida, como demonstrado nas atividades de "Sistema de Alarme" e "Jogo da Reciclagem" que utilizam a interação física para desencadear ações programadas.

COMO OBTER E COMUNIDADE

- **Download:** A Makey Makey é um hardware. Não há um "download" direto do dispositivo. No entanto, ele funciona "plug and play" na maioria dos sistemas operacionais (Windows, macOS, Chrome OS), sem a necessidade de instalar drivers específicos. Para interagir com ele, você pode usar softwares como Pictoblox, Scratch, ou qualquer aplicativo que responda a entradas de teclado e mouse.
- **IDEs (Ambientes de Desenvolvimento):** O Makey Makey não possui uma IDE própria. Ele é projetado para ser usado com outras plataformas de programação, especialmente aquelas que trabalham com blocos, como Scratch e Pictoblox, onde os eventos de "pressionar tecla" ou "clique de mouse" são programados.
- **MakeCode Arcade:** é um ambiente de programação online e gratuito, desenvolvido pela Microsoft, que permite a criação de jogos estilo arcade em 2D. Ele utiliza uma interface de programação por blocos, facilitando o aprendizado e a criação de jogos, mesmo para iniciantes. Os jogos criados podem ser jogados no navegador ou em consoles portáteis compatíveis com o MakeCode Arcade. É totalmente possível e divertido, usar a Makey Makey como controle para os jogos criados no MakeCode Arcade.



- **(Alguns) Links para exploração:**

- <https://makeymakey.com/>
- <https://makeymakey.com/pages/how-to>
- <https://makeymakey.com/pages/plug-and-play-makey-makey-apps>
- <https://www.youtube.com/watch?v=y0yoaeVno8c>
- https://www.youtube.com/watch?v=hbHFY2k_k4
- <https://www.youtube.com/watch?v=xdHdQ04goic>
- <https://learn.sparkfun.com/tutorials/makey-makey-classic-hookup-guide>
- <https://www.instructables.com/teachers/makeymakey/projects/>
- <https://www.blogdarobotica.com/2024/04/15/o-que-e-o-makey-makey/>
- <https://www.youtube.com/watch?v=VRmooxftByM>
- <https://www.youtube.com/watch?v=uS0o-JATnCU&list=PL0YzzcMQrJE8QXslnmL7GziL4C6Xhr2-j>
- <https://www.blogdarobotica.com/?s=makey+makey>
- <https://www.tinkercad.com/blog/explore-microbit-with-tinkercad>
- <https://www.makerhero.com/blog/8-jogos-com-makey-makey-kit/>

CAPÍTULO 3

CURSO DE FORMAÇÃO DOCENTE



MÃO NA MASSA - PROJETOS GUIADOS PARA AS AULAS DE MATEMÁTICA

Neste capítulo, a teoria se encontra com a prática. Após explorarmos as bases pedagógicas da cultura maker, construcionismo, Aprendizagem Baseada em Projetos (PBL) e Rotação por Estações, e nos familiarizarmos com as ferramentas — Pictoblox, Arduino Uno, Micro:bit e Makey Makey — é hora de colocar as "mãos na massa".

Aqui, apresentaremos uma série de projetos guiados, inspirados nas atividades desenvolvidas pelos professores no curso de formação. Cada projeto será detalhado passo a passo, desde a montagem do circuito até a programação em blocos no Pictoblox, sempre destacando os conceitos matemáticos envolvidos e as possibilidades de aplicação em sala de aula.

O objetivo é que estes projetos sirvam como ponto de partida e inspiração para que você, professor, possa replicá-los, adaptá-los e, o mais importante, criar suas próprias experiências de aprendizagem, transformando a matemática em algo tangível, interativo e divertido.

Os projetos estão organizados pelos encontros em que foram abordados no curso, seguindo a lógica das estações de trabalho e os problemas geradores temáticos.

ENCONTRO 1: “OLÁ MUNDO!”

O primeiro encontro do curso de formação teve como foco a apresentação inicial do curso, a aplicação da avaliação diagnóstica, uma palestra de base conceitual sobre os temas abordados no curso e, principalmente, a introdução prática às placas embarcadas e ao ambiente de programação em blocos Pictoblox.

A palestra intitulada: **“Pensamento Computacional e Robótica: Possibilidades para Práticas Pedagógicas Inovadoras”**, proferida pelo professor Marcus Maltempi, teve o objetivo de contribuir na formação da base teórica dos professores participantes do curso, além de ressaltar o histórico do avanço das pesquisas em robótica educacional e metodologias ativas, no âmbito da educação matemática, sempre ressaltando a importância de buscarmos acompanhar essa evolução, por meio da formação docente, inicial e continuada.



Além da palestra teórica, a prática de “mão na massa” com as placas, foi um momento crucial para familiarizar os professores com as ferramentas que seriam utilizadas nos projetos mais complexos dos encontros seguintes, funcionando como um "Olá Mundo!" da robótica educacional.



Projeto1: Acendendo um LED

Placas Utilizadas: Arduino Uno, Micro:bit, Makey Makey

Objetivo de Aprendizagem:

- **Matemático:** Compreender conceitos básicos de lógica (ligado/desligado), sequências simples e a relação de causa e efeito.
- **Computacional:** Desenvolver habilidades iniciais de programação em blocos (Pictoblox) para controlar atuadores digitais (LEDs), entender o conceito de entrada/saída digital e realizar a primeira interação entre hardware e software.

Conceitos Matemáticos Envolvidos:

- **Lógica Booleana:** Introdução intuitiva aos estados "verdadeiro" (ligado) e "falso" (desligado).
- **Sequência:** A ordem das instruções na programação para que o LED acenda e apague.
- **Contagem/Tempo (intuitiva):** Percepção de intervalos de tempo para o piscar do LED.

Competências e Habilidades da BNCC:

Estes experimentos introdutórios foram essenciais para os professores participantes do curso, pois serviram como base para que eles pudessem trabalhar com outros mais complexos a posteriori. Para alguns deles, era o primeiro contato com as placas embarcadas e o circuito eletrônico. À partir deles, destacamos, com base na BNCC:

- **Competências Gerais para o Ensino Fundamental:**
 - **2. Pensamento Científico, Crítico e Criativo:** Estimula a curiosidade sobre como a eletricidade e a programação podem controlar objetos.

- **5. Cultura Digital:** Primeiro contato prático com um microcontrolador e a ideia de "programar" um dispositivo físico.

- **Competências Específicas de Matemática:**

- **Álgebra:** Introdução à ideia de comando e resposta.

Para o Ensino Médio (foco em fundamentos):

- **Competências Gerais para o Ensino Médio:**

- **2. Pensamento Científico, Crítico e Criativo:** Consolida a compreensão de circuitos básicos e a lógica de programação.
- **5. Cultura Digital:** Reforça a base para projetos mais complexos, compreendendo a interação entre hardware e software.

- **Competências Específicas de Matemática e suas Tecnologias:**

- **Lógica:** Fundamentação para a lógica de programação e circuitos digitais.

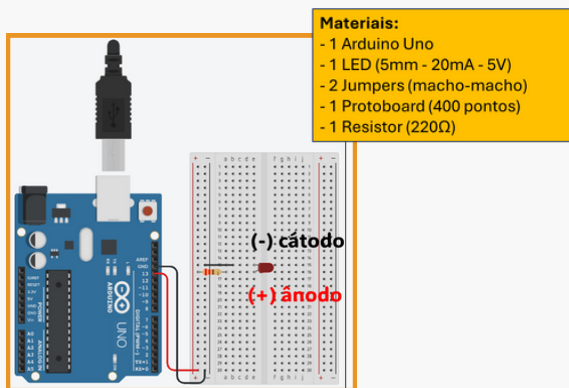
Materiais Necessários:

- Placa Arduino Uno
- Placa Micro:bit
- Placa Makey Makey
- Cabos USB para cada placa
- Computador com Pictoblox instalado
- Protoboard (placa de ensaio usada para construir e testar circuitos eletrônicos temporários sem a necessidade de solda)
- LEDs
- Resistores de 220 Ohms
- Fios jumpers (macho-macho, macho-fêmea)
- Cabos jacaré
- Objetos condutores

Passo a Passo da Montagem do circuito para cada placa:

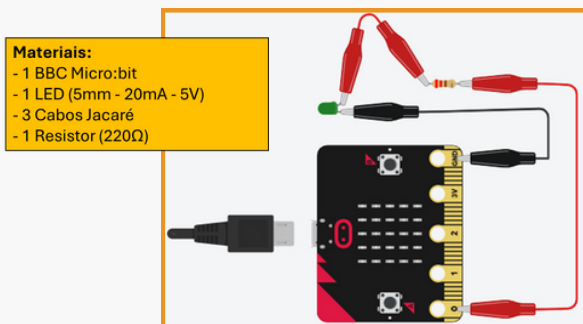
1. Para Arduino Uno:

- Conecte a perna mais longa (ânodo) do LED a um resistor de 220 Ohm.
- Conecte a outra ponta do resistor à porta digital 13 do Arduino Uno.
- Conecte a perna mais curta (catodo) do LED ao GND (terra) do Arduino.
- Conecte o cabo USB do Arduino Uno ao computador.



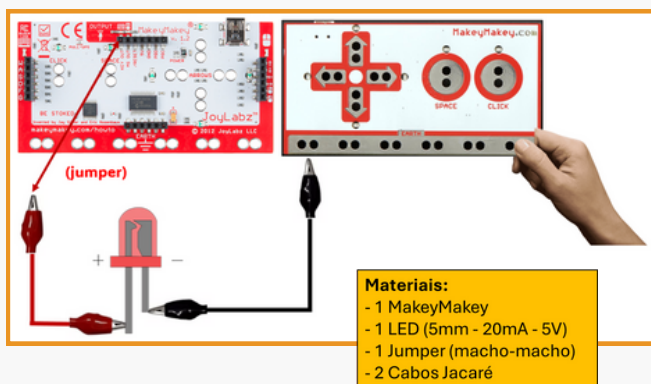
2. Para Micro:bit:

- Conecte a perna mais longa (ânodo) do LED a um resistor de 220 Ohm.
- Conecte a outra ponta do resistor à porta 0 da Micro:bit.
- Conecte a perna mais curta (catodo) do LED ao GND (terra) da Micro:bit.
- Conecte o cabo USB da Micro:bit ao computador.



3. Para Makey Makey:

- Conecte a perna mais longa (ânodo) do LED a um cabo jacaré.
- Conecte o cabo jacaré anterior à um cabo jumper macho-macho e este ao pino D14 (no verso da placa).
- Conecte a perna mais curta (catodo) do LED ao GND (terra) da Makey Makey.
- Conecte este cabo a algo que o usuário irá segurar (ex: a própria mão)
- Conecte o cabo USB do Makey Makey ao computador.



Passo a Passo da Programação no Pictoblox:

1. Abra o Pictoblox: Inicie o software Pictoblox em seu computador.

2. Selecione o Dispositivo:

- **Para Arduino Uno:** No menu "Dispositivos", selecione "Arduino" e "Arduino Uno". Conecte-o.
- **Para Micro:bit:** No menu "Dispositivos", selecione "Micro:bit". Conecte-o.
- **Para Makey Makey:** No menu "Dispositivos", selecione "Makey Makey".

3. Adicionar Extensões: Certifique-se de ter as extensões correspondentes ativadas para cada placa.

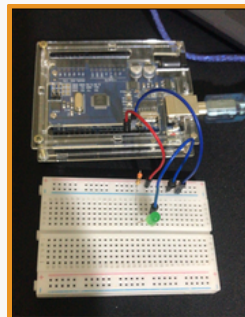
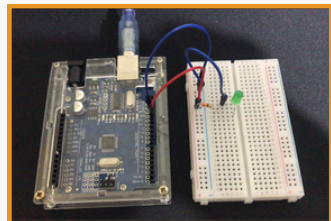
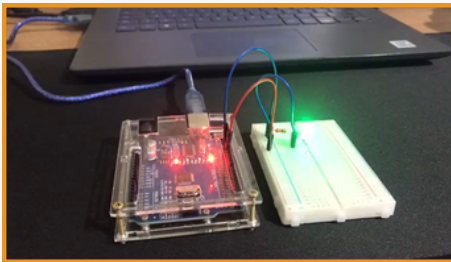
4. Programação "Olá Mundo!" para cada placa:

- **Arduino Uno (piscar LED):**

- Comece com o bloco de evento quando o Arduino Uno iniciar.
- Dentro deste, use um bloco sempre.
- Dentro do sempre:
 - definir o pino digital 13 como alto
 - esperar 1 segundo
 - desligar o pino digital 13 como baixo
 - esperar 1 segundos
 - opcional: caso queira tocar um som à cada ciclo de liga/desliga, insira o bloco “toque o som” no início ou final do loop.

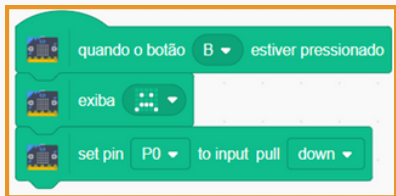
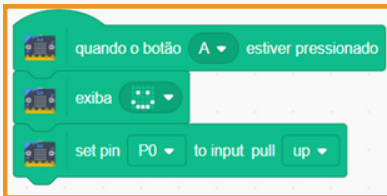


- **Na placa:**

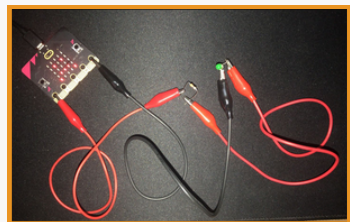
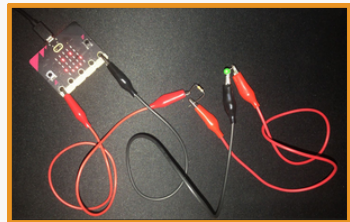
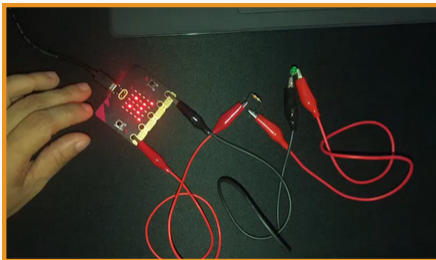


- **Micro:bit (mostrar ícone/texto):**

- Comece com o bloco de evento quando o botão "A" estiver pressionado, exiba o ícone "feliz".
- Defina o pino "P0" como "alto", isto é, ligado.
- Com o bloco de evento quando o botão "B" estiver pressionado, exiba o ícone "triste".
- Defina o pino "P0" como "baixo", isto é, desligado.
- Dessa forma, o LED será ligado (ou desligado) quando seus respectivos botões forem pressionados.



- **Na placa:**

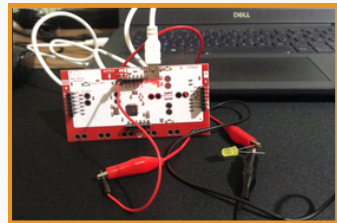
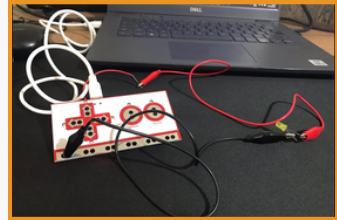
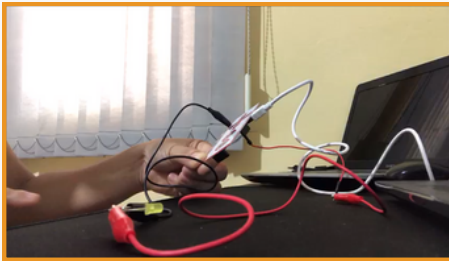


- **Makey Makey (acender um LED ao pressionar tecla):**

- Comece com o bloco de evento quando a tecla [espaço] for pressionada (ou tocada pelo usuário).
- Dentro deste:
 - tocar som [A Sax] (ou outro som da biblioteca).
 - dizer "LED LIGADO" (Sprite no Palco).
 - Após 1 segundo, dizer "LED DESLIGADO".



- **Na placa:**



Testando o Projeto:

- **Arduino Uno:** Após carregar o código, o LED conectado ao pino 13 do Arduino Uno deverá piscar a cada segundo.
- **Micro:bit:** Após transferir o código, a matriz de LEDs da Micro:bit deverá exibir o ícone alegre (ou triste) e ligar (desligar) o LED, a cada ciclo. Nesta programação, o LED só acenderá ou apagará, mediante o pressionar de seus respectivos botões "A" ou "B".
- **Makey Makey:** Com o cabo "terra" sendo segurado, somente quando tocar no cabo jacaré, conectado ao pino "espaço", um som será reproduzido, e o LED acenderá. Também, a mensagem "LED LIGADO" aparecerá no palco do Pictoblox por 1 segundo.

Sugestões de Aplicação em Sala de Aula de Matemática:

- **Introdução à Programação e Lógica:** Usar esses exemplos para explicar o que é um algoritmo, uma sequência de instruções e a lógica de "ligar/desligar".
- **Contagem e Padrões:** Observar a frequência do piscar do LED ou a repetição do ícone na Micro:bit.
- **Relação Causa e Efeito:** Demonstrar como uma ação física (tocar um objeto com Makey Makey) gera uma resposta digital (som/mensagem).

Desafios/Variações:

- **Arduino Uno:** Variar o tempo de piscar do LED. Adicionar mais LEDs e fazê-los piscar em padrões diferentes.
- **Micro:bit:** Criar animações simples na matriz de LEDs. Exibir números em sequência.
- **Makey Makey:** Conectar diferentes objetos a outras teclas e criar um "piano" simples com sons e cores variados.



ENCONTRO 2: CIDADES INTELIGENTES

O segundo encontro do curso de formação focou no tema "Cidades Inteligentes", desafiando os professores a pensarem em soluções tecnológicas para otimizar aspectos do ambiente urbano. Os subproblemas propostos em cada estação foram o ponto de partida para a construção de protótipos funcionais com diferentes placas.

Projeto 2.1: Controle de Iluminação e Segurança

Placa Utilizada: Arduino Uno

Objetivo de Aprendizagem:

- **Matemático:** Compreender conceitos de proporcionalidade, grandezas (intensidade luminosa), lógica booleana (ligado/desligado), e noções básicas de dados.
- **Computacional:** Desenvolver habilidades de programação em blocos (Pictoblox) para leitura de sensores analógicos, controle de atuadores digitais (LEDs) e uso de estruturas condicionais (se/então/senão). Entender o conceito de entrada/saída de dados em um microcontrolador.

Conceitos Matemáticos Envolvidos:

- **Proporcionalidade:** A intensidade de luz lida pelo sensor afeta diretamente a resposta do sistema.
- **Inequações/Limites:** Definição de um valor limite para acionar ou desativar o sistema (ex: se a luz estiver abaixo de X, ligar o LED).
- **Lógica Booleana:** Operações verdadeiro/falso na tomada de decisão (luz baixa = VERDADEIRO, luz alta = FALSO).
- **Dados e Medição:** Coleta de valores contínuos de luz e interpretação desses dados.

Competências e Habilidades da BNCC:

Este projeto, centrado no uso do Arduino Uno, é particularmente relevante para o Ensino Médio, mas pode ser adaptado para o Ensino Fundamental com simplificações.

Para o Ensino Médio:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Estimula a investigação, a experimentação e a análise de fenômenos naturais (luz), aplicando conhecimentos científicos e tecnológicos para propor soluções.
- **5. Cultura Digital:** Promove a compreensão e o uso da tecnologia digital (hardware e software) de forma crítica, significativa e ética para resolver problemas complexos, como a automação de sistemas.
- **7. Argumentação:** Desenvolve a capacidade de formular e defender ideias sobre o funcionamento do sistema, justificar as escolhas de programação e os limites de atuação do sensor.
- **8. Autoconhecimento e Autocuidado:** O processo de depuração e ajuste do sistema, com tentativa e erro, desenvolve a resiliência e a capacidade de lidar com desafios.
- **10. Responsabilidade e Cidadania:** Contribui para a reflexão sobre o uso eficiente de recursos (energia) e a segurança em ambientes urbanos, um dos desafios das "Cidades Inteligentes".

- **Competências Específicas de Matemática e suas Tecnologias:**

- **Habilidades Relacionadas a Grandezas e Medidas:**
 - Analisar e interpretar dados obtidos de sensores para modelar situações reais, utilizando diferentes escalas de valores (por exemplo, 0 a 1023 no LDR).

- Resolver problemas que envolvam grandezas diretamente proporcionais (intensidade luminosa e valor lido pelo sensor), estabelecendo relações entre elas.
- **Habilidades Relacionadas a Álgebra:**
 - Modelar fenômenos por meio de funções (ainda que implícita, a relação entre luz e valor do sensor é uma função), compreendendo o comportamento e a aplicação de limites para tomada de decisão no código.
- **Habilidades Relacionadas à Probabilidade e Estatística:**
 - Interpretar e analisar dados brutos do sensor, compreendendo a variabilidade e a necessidade de calibração para garantir o funcionamento adequado do sistema.

Para o Ensino Fundamental:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Exploração de fenômenos simples de luz e sombra, e como a tecnologia pode "perceber" e "reagir" a eles.
- **5. Cultura Digital:** Primeiro contato com a ideia de programar um dispositivo físico e entender a relação entre comando (código) e ação (LED acender/apagar).

- **Competências Específicas de Matemática:**

- **Grandezas e Medidas:** Compreensão intuitiva de "mais luz" e "menos luz", e como a tecnologia pode medir isso.
- **Álgebra:** Introdução à ideia de "se/então" para tomar decisões baseadas em uma condição simples (luz alta ou baixa).
- **Probabilidade e Estatística:** Coleta de alguns dados de luminosidade em diferentes momentos e a observação de padrões (ex: mais claro de dia, mais escuro à noite).

Materiais Necessários:

- 1 Arduino Uno
- 1 Protoboard (400 pontos)
- 1 LED (5mm - 20mA - 5V)
- 1 Resistores 220Ω (ohms)
- 1 Resistores 300Ω (ohms)
- 5 Jumpers (Macho x Macho)
- 1 Sensor de Luminosidade (LDR)

Passo a Passo da Montagem do Circuito:

1. Montagem do Circuito do LDR (Sensor de Luz):

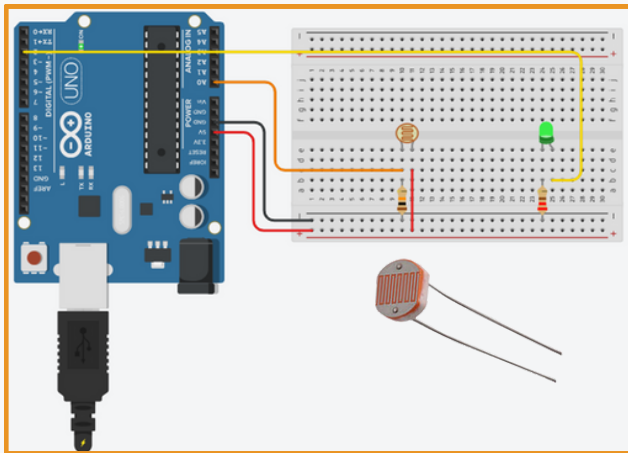
- Conecte uma perna do LDR à porta analógica A0 do Arduino Uno.
- Conecte a outra perna do LDR a uma linha de 5V na protoboard.
- Conecte um resistor de 10k Ohm da mesma perna do LDR que está conectada à A0 até o GND (terra) na protoboard.
- Certifique-se de que a linha de 5V na protoboard esteja conectada ao 5V do Arduino Uno e a linha de GND na protoboard esteja conectada ao GND do Arduino Uno. Este arranjo forma um divisor de tensão, permitindo ao Arduino Uno ler a variação da resistência do LDR como uma tensão.

2. Montagem do Circuito dos LEDs:

- **LED Verde (Iluminação Automática):** Conecte a perna mais longa (ânodo) do LED verde a um resistor de 220 Ohm. Conecte a outra ponta do resistor à porta digital 9 do Arduino Uno. Conecte a perna mais curta (catodo) do LED verde ao GND na protoboard.
- **LED Vermelho (Alerta de Segurança - opcional, para simular um alarme):** Conecte a perna mais longa (ânodo) do LED vermelho a um resistor de 220 Ohm. Conecte a outra ponta do resistor à porta digital 10 do Arduino Uno. Conecte a perna mais curta (catodo) do LED vermelho ao GND na protoboard.

3. Conexão Final:

- Verifique todas as conexões para garantir que estão firmes e corretas.
- Conecte o cabo USB do Arduino Uno ao seu computador.



Passo a Passo da Programação no Pictoblox:

1. Abra o Pictoblox: Inicie o software Pictoblox em seu computador.

2. Selecione o Dispositivo Arduino: No menu "Dispositivos" (Devices), selecione "Arduino" e, em seguida, "Arduino Uno" (ou o modelo que estiver usando). Conecte-o via USB ao computador.

3. Adicionar Extensões: Para interagir com o Arduino Uno, você precisará adicionar as extensões de "Arduino" e, opcionalmente, de "Display" para visualizar valores.

4. Estrutura Básica:

- Comece com o bloco de evento quando o Arduino Uno iniciar.
- Dentro deste, use um bloco sempre para que o programa monitore continuamente o sensor de luz.

5. Leitura do Sensor LDR: Use o bloco ler pino analógico A0 para obter o valor da luminosidade. Os valores lidos pelo pino analógico variam de 0 (muita luz, baixa resistência do LDR) a 1023 (pouca luz, alta resistência do LDR).

6. Controle da Iluminação (LED Verde):

- Utilize um bloco se/então/senão.
- Na condição, use um operador de comparação. Por exemplo: se (ler pino analógico A0) < 70 (este valor 70 é um exemplo de "limiar" e pode ser ajustado na prática).
 - Então: ligar o pino digital 2 (para o LED verde). Isso significa que se a luminosidade estiver abaixo do limiar (o valor lido for alto, indicando pouca luz), a iluminação automática será ativada.
 - Senão: desligar o pino digital 2. Se a luz ambiente for suficiente, o LED verde ficará apagado.

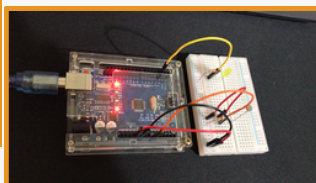
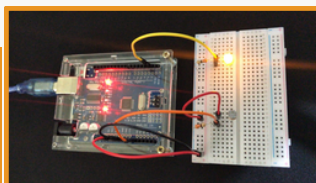
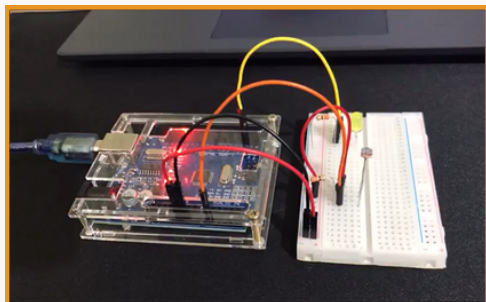
7. Alerta de Segurança (LED Vermelho - Opcional):

- Adicione outro bloco se/então/senão.
- Na condição, use um operador de comparação diferente para um cenário de segurança. Por exemplo: se (ler pino analógico A0) > 100 (indicando que há MUITA luz, talvez uma lanterna apontada para o sensor, simulando uma quebra de segurança em um ambiente escuro que deveria estar).
 - Então: ligar o pino digital 3 (para o LED vermelho), e talvez adicionar um som de alerta.
 - Senão: desligar o pino digital 3.

```
quando for clicado
sempre
diga junto Luminosidade do ambiente: com ler sensor analógico luz / fotoresistor no pino A0
mude LUZ para ler sensor analógico luz / fotoresistor no pino A0
se LUZ < 70 então
  Defnir pino digital 2 saída como ALTO
senão
  Defnir pino digital 2 saída como BAIXO
```



Na Placa:



Testando o Projeto:

Após carregar o código para o Arduino Uno (clcando no botão de upload no Pictoblox), o sistema começará a funcionar.

- Quando o ambiente estiver escuro (ou você cobrir o LDR com a mão), o LED amarelo deverá acender, simulando a iluminação automática.
- Quando o ambiente estiver claro, o LED amarelo deverá apagar.
- Se você adicionar o LED vermelho, ao apontar uma fonte de luz forte (como a lanterna do celular) diretamente para o LDR em um ambiente já iluminado (ou não), simulando uma anomalia, o LED vermelho deverá acender.
- Você pode monitorar os valores do LDR em tempo real no Pictoblox usando o "Monitor Serial" ou exibindo o valor em um Sprite.

Sugestões de Aplicação em Sala de Aula de Matemática:

- **Gráficos de Linha:** Coletar dados de luminosidade ao longo do dia em diferentes pontos da escola e construir gráficos de linha para analisar padrões (horas mais escuras/claras).

- **Limites e Proporcionalidade:** Explorar como a mudança do "threshold" (o valor de 400 no exemplo) afeta o comportamento do sistema. Discutir o que significa o sensor ler um valor de X, Y, Z e como isso se relaciona com a intensidade da luz.
- **Otimização de Recursos:** Calcular a economia de energia ao usar um sistema de iluminação inteligente versus um sistema que permanece ligado o tempo todo.
- **Introdução à Lógica Booleana:** Utilizar o projeto para explicar o conceito de condições "verdadeiro/falso" e como elas controlam o fluxo de um programa.
- **Interpretação de Dados Analógicos:** Debater por que o sensor LDR fornece uma faixa de valores (0-1023) e não apenas "luz" ou "não luz", e como essa "granularidade" é útil.

Desafios/Variações:

- Adicionar um buzzer para emitir um som quando o LED vermelho de alerta acender.
- Integrar um sensor de presença (PIR) para que a luz só acenda se estiver escuro E houver movimento.
- Utilizar um display LCD para mostrar o valor exato da luminosidade e o estado do sistema ("Luz Ligada", "Luz Desligada", "Alerta!").
- Desenvolver um protótipo de "semáforo inteligente" que muda de cor de acordo com a luminosidade ambiente (mais luz = semáforo fica ligado, menos luz = semáforo apaga).

Projeto 2.2: Controle de Iluminação com Sensor de Presença (PIR)

Placa Utilizada: Arduino Uno

Objetivo de Aprendizagem:

- **Matemático:** Compreender conceitos de lógica booleana (presença/ausência), contagem de eventos, duração de tempo e noções de área de detecção.
- **Computacional:** Desenvolver habilidades de programação em blocos (Pictoblox) para leitura de sensores digitais, controle de atuadores digitais (LEDs) e uso de estruturas condicionais e temporizadores.

Conceitos Matemáticos Envolvidos:

- **Lógica Booleana:** Operações verdadeiro/falso para detecção de movimento (movimento = VERDADEIRO, sem movimento = FALSO).
- **Contagem:** Contagem de eventos (quantas vezes o movimento foi detectado ou a luz foi acesa).
- **Tempo e Duração:** Definição de intervalos de tempo para a luz permanecer acesa após a detecção de movimento.
- **Geometria (intuitiva):** Compreensão da área de cobertura do sensor e como a posição afeta a detecção.

Competências e Habilidades da BNCC:

Este projeto, centrado no uso do Arduino Uno com sensor PIR, é relevante para o Ensino Fundamental e o Ensino Médio, podendo ser adaptado sem complexidade para outros níveis de ensino.

Para o Ensino Médio:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Estimula a investigação de fenômenos de detecção, a experimentação com limites de alcance do sensor e a aplicação de lógica para automação.
 - **5. Cultura Digital:** Promove a compreensão e o uso da tecnologia digital (hardware e software) para criar sistemas automatizados que otimizam recursos e aumentam a segurança.
 - **7. Argumentação:** Desenvolve a capacidade de justificar as escolhas de tempo de ativação da luz e a eficácia do sistema em diferentes cenários.
 - **10. Responsabilidade e Cidadania:** Contribui para a reflexão sobre a economia de energia em edifícios inteligentes e a aplicação de sistemas de segurança.
- **Competências Específicas de Matemática e suas Tecnologias:**
 - **Habilidades Relacionadas a Grandezas e Medidas:**
 - Analisar e interpretar dados de tempo (duração da luz acesa) e relacioná-los com a eficiência do sistema.
 - **Habilidades Relacionadas a Álgebra:**
 - Modelar a lógica de ativação/desativação da luz por meio de funções condicionais, compreendendo o comportamento do sistema com base na entrada do sensor.
 - **Habilidades Relacionadas à Probabilidade e Estatística:**
 - Coletar dados sobre a frequência de detecções em um período e analisar padrões de uso do ambiente para otimizar o sistema.

Para o Ensino Fundamental:

- **Competências Gerais:**
 - **2. Pensamento Científico, Crítico e Criativo:** Exploração da ideia de "perceber" movimento e "reagir"

- a ele, desenvolvendo a curiosidade sobre como a tecnologia funciona.
- **5. Cultura Digital:** Primeiro contato com a programação de um sensor de movimento para controlar um dispositivo físico (luz).

- **Competências Específicas de Matemática:**

- **Lógica:** Introdução à ideia de "se/então" para acender a luz apenas quando há movimento.
- **Contagem:** Contar quantas vezes a luz acendeu em um determinado período.
- **Grandezas e Medidas:** Compreensão intuitiva de "tempo" que a luz permanece acesa.

Materiais Necessários:

- 1 Arduino Uno
- 1 Protoboard (400 pontos)
- 1 LED (5mm - 20mA - 5V)
- 1 Resistor 220Ω (ohms)
- 7 Jumpers (4 Macho x Macho e 3 Fêmea x Macho)
- 1 Sensor de Presença (PIR DYP-ME003)

Passo a Passo da Montagem do Circuito:

1. Montagem do Sensor PIR:

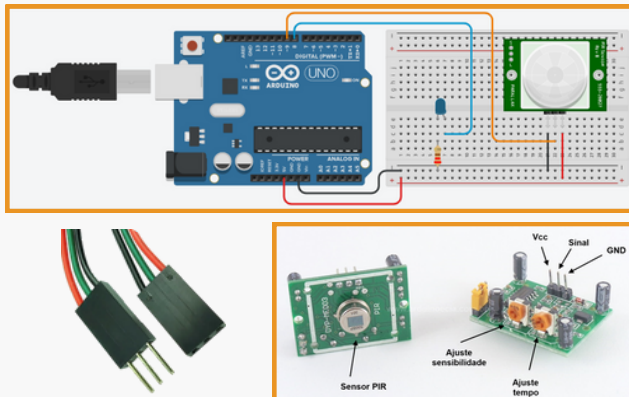
- O sensor PIR geralmente possui 3 pinos: VCC (ou +), GND (ou -) e SINAL (saída de dados).
- Conecte o pino VCC do PIR ao 5V do Arduino Uno.
- Conecte o pino GND do PIR ao GND do Arduino Uno.
- Conecte o pino SINAL do PIR a uma porta digital do Arduino Uno, por exemplo, pino digital 9.

2. Montagem do Circuito dos LEDs:

- Conecte a perna mais longa (ânodo) do LED a um resistor de 220 Ohm. Conecte a outra ponta do resistor à porta digital 8 do Arduino Uno. Conecte a perna mais curta (catodo) do LED ao GND na protoboard.

3. Conexão Final:

- Verifique todas as conexões para garantir que estão firmes e corretas.
- Conecte o cabo USB do Arduino Uno ao seu computador.



Passo a Passo da Programação no Pictoblox:

1. Abra o Pictoblox: Inicie o software Pictoblox em seu computador.

2. Selecione o Dispositivo Arduino: No menu "Dispositivos", selecione "Arduino" e, em seguida, "Arduino Uno" (ou o modelo que estiver usando). Conecte-o via USB ao computador.

3. Adicionar Extensões: Para interagir com o Arduino, você precisará adicionar as extensões de "Arduino" e, opcionalmente, de "Display" para visualizar valores.

4. Estrutura Básica:

- Comece com o bloco de evento quando o Arduino Uno iniciar.
- Dentro deste, use um bloco sempre para que o programa monitore continuamente o sensor PIR.

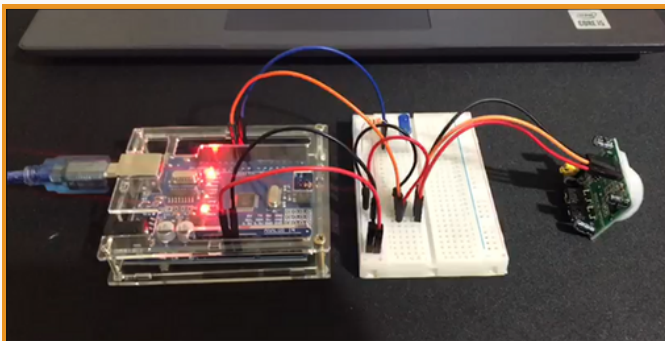
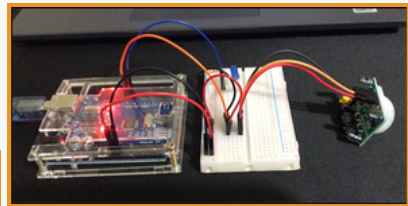
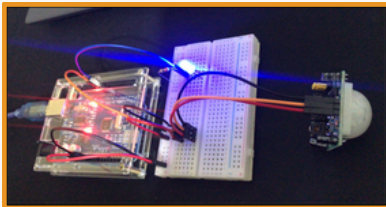
5. Leitura do Sensor PIR:

- Utilize um bloco se/então/senão.
- Na condição, use o bloco ler pino digital [9] e compare com ALTO (HIGH). Por exemplo: se (ler pino digital 9) = ALTO.

- **Então:** ligar o pino digital 8 (para o LED). Isso significa que se movimento for detectado, a luz acenderá.
- Adicione um bloco **“esperar 2 segundos”** para que a luz permaneça acesa por um tempo após a última detecção de movimento.
- **Senão:** desligar o pino digital 8. Se não houver movimento, o LED ficará apagado.

```
quando for clicado
sempre
se ler sensor digital PIR no pino 9 então
  Definir pino digital 8 saída como ALTO
  espere 2 seg
senão
  Definir pino digital 8 saída como BAIXO
```

Na Placa:



Testando o Projeto:

Após carregar o código para o Arduino Uno (clique no botão de upload no Pictoblox), o sistema começará a funcionar.

- Quando você se mover na frente do sensor PIR, o LED deverá acender.
- O LED permanecerá aceso pelo tempo definido no bloco esperar após a última detecção de movimento.
- Quando não houver mais movimento na área de detecção do sensor e o tempo de espera terminar, o LED deverá apagar.

Você pode monitorar o estado do sensor PIR em tempo real no Pictoblox usando o "Monitor Serial" ou exibindo o valor em um Sprite.

Sugestões de Aplicação em Sala de Aula de Matemática:

- **Contagem de Pessoas/Eventos:** Utilizar o sensor para contar quantas vezes uma pessoa passa por um local em um determinado período. Construir tabelas e gráficos de frequência.
- **Cálculo de Tempo:** Medir o tempo que a luz permanece acesa e calcular o consumo de energia em diferentes cenários de uso.
- **Otimização de Consumo:** Discutir como o ajuste do tempo de permanência da luz acesa pode otimizar o consumo de energia em espaços como corredores ou banheiros.
- **Introdução a Sistemas de Segurança:** Simular um sistema de alarme simples, onde a detecção de movimento aciona um alerta (além da luz, um som).
- **Análise de Dados de Movimento:** Coletar dados sobre os horários de maior e menor movimento em um ambiente escolar e analisar esses padrões.

Desafios/Variações:

- Adicionar um buzzer para emitir um som quando o movimento for detectado.
- Contar e exibir o número total de detecções de movimento em um display LCD ou no Palco do Pictoblox.
- Variar o tempo de ativação da luz com base em outras condições (ex: período do dia, nível de luz ambiente).
- Integrar com um sensor de luz (LDR) para que a luz só acenda se estiver escuro e houver movimento.
- Desenvolver um sistema de "presença inteligente" que registra a entrada e saída de pessoas.

Projeto 2.3: Controle de Tráfego

Placa Utilizada: Micro:bit

Objetivo de Aprendizagem:

- **Matemático:** Explorar conceitos de sequência, tempo, contagem, lógica condicional (se/então/senão) e noções básicas de otimização de fluxo.
- **Computacional:** Desenvolver habilidades de programação em blocos (Pictoblox) para controlar a matriz de LEDs, utilizar temporizadores e blocos de controle de fluxo (loops e condicionais).

Conceitos Matemáticos Envolvidos:

- **Sequência e Padrões:** A ordem das luzes do semáforo e a repetição do ciclo.
- **Tempo e Duração:** Definição de intervalos de tempo para cada estado do semáforo.
- **Contagem:** Possibilidade de incluir um contador de veículos (simulado por botões) para adaptar o tempo do semáforo.
- **Lógica Condicional:** Decisões sobre quando mudar o estado do semáforo com base no tempo ou em outras entradas.
- **Otimização (intuitiva):** Discussão sobre como a mudança nos tempos de ciclo pode afetar o fluxo de tráfego.

Competências e Habilidades da BNCC:

Este projeto, com a Micro:bit, é extremamente versátil e relevante para o Ensino Fundamental e o Ensino Médio.

Para o Ensino Fundamental:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Observar e descrever sequências, formular hipóteses sobre o funcionamento de um semáforo e criar um modelo simplificado.
 - **4. Comunicação:** Expressar ideias sobre o projeto e o funcionamento do semáforo através da representação visual na matriz de LEDs.
 - **5. Cultura Digital:** Primeiro contato com a programação de um dispositivo físico e a visualização imediata do impacto do código.
 - **9. Empatia e Cooperação:** Trabalhar em grupo para desenvolver a lógica do semáforo e testar seu funcionamento.
 - **10. Responsabilidade e Cidadania:** Discutir a importância dos semáforos para a organização do tráfego e a segurança nas cidades.
- **Competências Específicas de Matemática:**
 - **Números:** Contagem de tempo (em segundos), compreensão de ordem numérica.
 - **Álgebra:** Introdução à ideia de "se... então..." para controlar a sequência das luzes do semáforo (ex: se for verde, então depois de X segundos, muda para amarelo).
 - **Geometria:** Reconhecimento de padrões visuais na matriz de LEDs.
 - **Probabilidade e Estatística:** Discussão sobre a aleatoriedade de um semáforo manual (se não houvesse programação) versus a previsibilidade de um semáforo programado.

Para o Ensino Médio:

- **Competências Gerais:**
 - **2. Pensamento Científico, Crítico e Criativo:** Modelagem de sistemas complexos (tráfego urbano)

com análise de variáveis (tempo de ciclo, fluxo de veículos), simplificações e proposição de soluções algorítmicas.

- **5. Cultura Digital:** Uso de plataforma de programação em blocos e hardware para criar um sistema de controle, explorando a relação entre algoritmo e comportamento do sistema.
 - **6. Trabalho e Projeto de Vida:** Planejamento e execução de um projeto, gerenciamento de tempo e recursos (simulados).
 - **7. Argumentação:** Justificar as escolhas de tempo para cada fase do semáforo e como isso impacta a otimização do fluxo.
 - **10. Responsabilidade e Cidadania:** Aprofundar a discussão sobre mobilidade urbana, impactos ambientais (engarrafamentos) e o papel da tecnologia na solução de problemas sociais.
-
- **Competências Específicas de Matemática e suas Tecnologias:**
 - **Habilidades Relacionadas a Álgebra:**
 - Modelar sistemas dinâmicos (a sequência do semáforo) utilizando variáveis de tempo e condicionais complexas.
 - Analisar a relação entre as variáveis (tempos de verde, amarelo, vermelho) e o comportamento geral do sistema.
 - **Habilidades Relacionadas a Funções:**
 - Compreensão de funções discretas (estados do semáforo mudando em intervalos de tempo fixos).
 - **Habilidades Relacionadas à Probabilidade e Estatística:**
 - Simular (com o uso de botões) e analisar dados de fluxo de veículos, discutindo como esses dados poderiam ser usados para otimizar os tempos do semáforo em um cenário real.

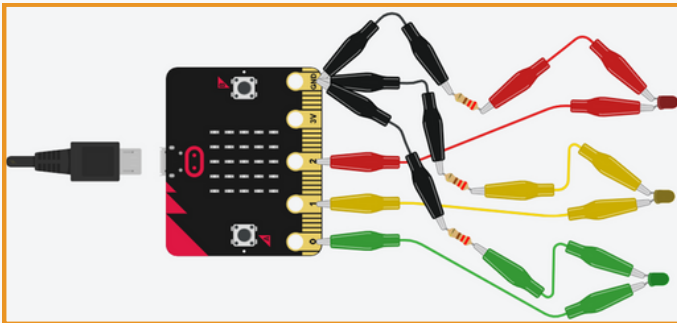
- Discussão de algoritmos de otimização de fila e tempo de espera.

Materiais Necessários:

- 1 BBC Micro:bit
- 3 LEDs (ou 1 Módulo Semáforo)
- 3 Resistores 220Ω (opcional pois a voltagem da micro:bit é de 3.3 volts, compatível com a do LED)
- 6 Cabos Jacaré

Passo a Passo da Montagem do Circuito:

A beleza da Micro:bit para este projeto é que ela possui todos os componentes necessários integrados. A montagem do circuito externo é muito simples. Basta conectar aos pinos 0, 1 e 2 da Micro:bit, cabos jacarés, resistores e os LEDs nas cores verde, amarelo e vermelho, conforme circuito mostrado abaixo. Depois, conecte a Micro:bit ao computador via cabo USB, e transfira para ela a programação feita.



Passo a Passo da Programação no Pictoblox:

- 1. Abra o Pictoblox:** Inicie o software Pictoblox em seu computador.
- 2. Selecione o Dispositivo Micro:bit:** No menu "Dispositivos", selecione "Micro:bit". Conecte-o via USB ao computador.
- 3. Adicionar Extensões:** Certifique-se de ter a extensão "Micro:bit" ativada.

4. Estrutura Básica:

- Comece com o bloco de evento quando a Micro:bit iniciar.
- Dentro deste, use um bloco sempre para que a programação dos LEDs seja executada continuamente, simulando o ciclo do semáforo.

5. Ciclo do Semáforo (Verde -> Amarelo -> Vermelho):

- **LED Verde:**
 - Crie a variável SIGA, para que possamos temporizá-la na matriz de LEDs da Micro:bit.
 - Defina o pino digital P0 como alto, para acender o LED verde.
 - Use o bloco display text para imprimir o texto “SIGA” na matriz de LEDs por 1 segundo.
 - Use o bloco exiba para acender um ícone na matriz de LED. Neste caso, usamos o “boneco palito” para simular a caminhada.
 - Use o bloco esperar 1 segundos (tempo de exibição do ícone “boneco palito”).
 - Use o bloco mude SIGA para 5, para que seja iniciada e mostrada a temporização na matriz de LEDs.
 - Use o bloco repita 5 vezes, para que, a cada segundo, seja diminuído em uma unidade, a temporização do visor, simulando o tempo permitido para a travessia do pedestre.
 - Após 5 segundos, o LED verde será apagado, isto é, o pino P0 será definido como baixo.
- Para os **LEDs Amarelo e Vermelho**, utilize a mesma programação acima, sempre observando o Pino digital correspondente, a criação das variáveis, a temporização dela e o ícone correspondente da ação indicada na matriz de LEDs da micro:bit.
- **Apagar LEDs:** Após cada estado, ou no início de cada novo estado, você pode usar o bloco limpar tela (ou apagar LEDs) para garantir que apenas o padrão desejado esteja visível.

```

quando for clicado
sempre
  set P0 Digital High
  display text SIGA delay 100 ms
  exiba
  espere 1 seg
  mude SIGA para 5
  repita 5 vezes
    display text SIGA delay 100 ms
  adicione -1 a SIGA
  espere 1 seg
  set P0 Digital Low

```

```

set P2 Digital High
display text PARE delay 100 ms
exiba
espere 1 seg
mude PARE para 4
repita 4 vezes
  display text PARE delay 100 ms
  adicione -1 a PARE
  espere 1 seg
set P2 Digital Low

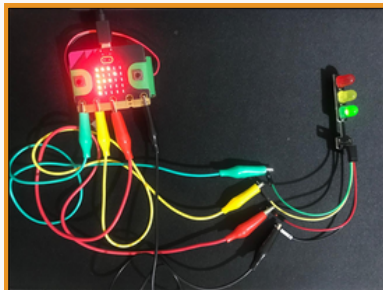
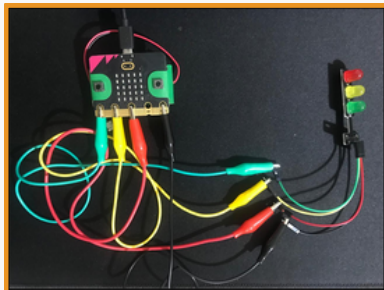
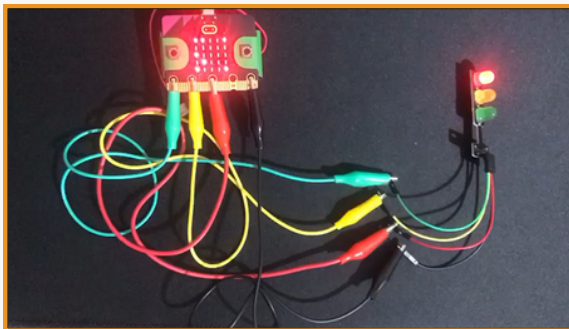
```

```

set P1 Digital High
display text ATENCAO delay 100 ms
exiba
espere 1 seg
mude ATENCAO para 3
repita 3 vezes
  display text ATENCAO delay 100 ms
  adicione -1 a ATENCAO
  espere 1 seg
set P1 Digital Low

```

Na Placa:



Adicionando Interatividade (Opcional - para Ensino Médio):

- Para simular um "sensor de tráfego" e otimização, você pode usar os botões A e B.
- Exemplo: quando o botão A for pressionado: aumentar a variável 'contador_veiculos' por 1.
- Use se/então para que, se contador_veiculos atingir um certo valor, o tempo do LED verde seja ajustado, ou o semáforo mude mais rapidamente para o vermelho. (Isso introduz a ideia de algoritmos adaptativos).
- Você pode exibir o contador_veiculos na matriz de LEDs usando mostrar número contador_veiculos.

Testando o Projeto:

Após transferir o código para a Micro:bit, ela começará a exibir uma sequência de ícones e temporizador do semáforo em sua matriz de LEDs, além do acendimento dos LEDs do circuito de forma contínua e sincronizada com a programação.

- Observe a transição de ícones, acendimento dos LEDs e os tempos definidos.
- Se implementou a parte opcional com os botões, pressione o botão A algumas vezes e observe se o semáforo reage de alguma forma (se você programou para isso).

Sugestões de Aplicação em Sala de Aula de Matemática:

- **Sequências Numéricas e Lógicas:** Analisar a sequência de cores e tempos do semáforo como um padrão. Criar semáforos com sequências mais complexas.
- **Medidas de Tempo:** Cronometrar os diferentes estados do semáforo e discutir a importância da precisão no tempo. Converter segundos para outras unidades.
- **Proporcionalidade e Frações:** Se um ciclo total dura 12 segundos (5s verde, 2s amarelo, 5s vermelho), que fração do tempo o semáforo fica verde? E amarelo?

- **Otimização e Algoritmos:** Desafiar os alunos a otimizar os tempos do semáforo para diferentes fluxos de tráfego (simulados). Como garantir que os carros não esperem muito? Qual é o tempo ideal?
- **Modelagem Matemática:** Criar um modelo matemático simples para prever o tempo total de um ciclo de semáforo com base nos tempos de cada fase.
- **Gráficos de Linha/Barra:** Coletar dados sobre o "fluxo de veículos" (número de vezes que o botão A foi pressionado em um período) e criar gráficos para visualizar o movimento.

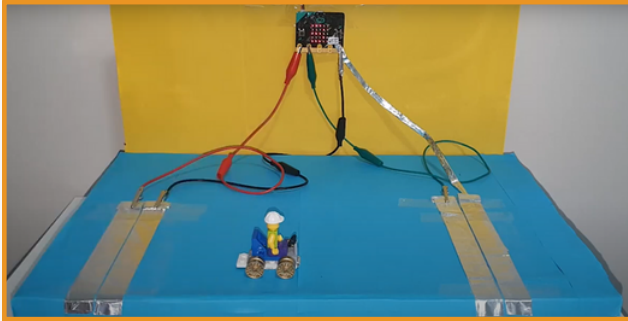
Desafios/Variações:

- Adicionar um segundo semáforo (simulado por outro conjunto de LEDs ou um Sprite no Pictoblox) para controlar o tráfego em uma interseção, garantindo que nunca haja duas luzes verdes ao mesmo tempo em direções que se cruzam (lógica mais complexa).
- Usar o sensor de luz da Micro:bit para que o semáforo funcione apenas durante o dia ou à noite (simulando um semáforo inteligente que se adapta à luminosidade).
- Integrar um sensor de movimento externo (se a Micro:bit estiver em uma protoboard) para detectar a "chegada de um veículo" e ajustar o tempo do semáforo dinamicamente.
- Criar um "controle de pedestres" usando o botão B, que aciona o sinal de pedestres e pausa o tráfego.

Remixando:

Segue abaixo a sugestão de dois projetos, que tratam da "LOMBADA ELETRÔNICA", disponíveis na web, e que remixamos durante o curso de formação docente. Ambos necessitam de outros materiais como caixa de papelão, fita adesiva (ou papel alumínio) e carrinhos de brinquedo (tipo Hot Wheels), para dar mais vida e ludicidade ao projeto.

- **Lombada Eletrônica 1:** Atividade Maker 8 - Lombada Eletrônica
 - <https://www.youtube.com/watch?v=myxgxOD77OM>



- **Lombada Eletrônica 2:** Mostra Micro:bit - Projeto Lombada Eletrônica - C.E. Padre Claudio Morelli
 - <https://www.youtube.com/watch?v=myxgxOD77OM>



Projeto 2.4: Sistema de Alarme

Placa Utilizada: Makey Makey

Objetivo de Aprendizagem:

- **Matemático:** Compreender conceitos de circuito aberto/fechado, lógica booleana (acionado/não acionado), sequências de eventos e introdução intuitiva a funções (entrada -> processamento -> saída).
- **Computacional:** Desenvolver habilidades de programação em blocos (no Pictoblox) para detectar eventos de teclado/mouse (via Makey Makey), reproduzir sons, exibir mensagens e criar interações básicas.

Conceitos Matemáticos Envolvidos:

- **Lógica Booleana:** O alarme é ativado se uma condição (circuito fechado) é verdadeira.
- **Sequências de Eventos:** A ordem em que o alarme dispara (som, mensagem, etc.).
- **Contagem:** Possibilidade de contar o número de vezes que o alarme foi acionado.
- **Geometria:** Criação de áreas "sensíveis" que, quando tocadas, ativam o alarme.

Competências e Habilidades da BNCC:

Este projeto, com o Makey Makey, é particularmente adequado para o Ensino Fundamental devido à sua natureza lúdica e concreta, mas pode ser aprofundado para o Ensino Médio com extensões de lógica e análise.

Para o Ensino Fundamental:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Explorar como diferentes materiais conduzem eletricidade para criar um "sensor" de alarme. Formular hipóteses sobre o que pode acionar o alarme.
 - **4. Comunicação:** Expressar ideias sobre o funcionamento do alarme e como ele pode ser usado para proteger algo.
 - **5. Cultura Digital:** Entender a relação entre um objeto físico, o Makey Makey e um programa de computador, transformando algo do cotidiano em um controle digital.
 - **9. Empatia e Cooperação:** Trabalhar em grupo para testar diferentes materiais e construir um sistema de alarme eficaz.
 - **10. Responsabilidade e Cidadania:** Discutir a importância da segurança e como a tecnologia pode ajudar a proteger bens.
- **Competências Específicas de Matemática:**
 - **Álgebra:** Introdução à lógica de "se... então..." para a ativação do alarme (ex: se o fio for tocado, então faça barulho).
 - **Grandezas e Medidas:** Compreensão de que um "toque" é um evento discreto que aciona uma resposta.
 - **Geometria:** Posicionamento de "sensores" em uma área para criar uma zona de proteção.

Para o Ensino Médio:

- **Competências Gerais:**
 - **2. Pensamento Científico, Crítico e Criativo:** Analisar a eficiência de diferentes materiais condutores para o alarme. Desenvolver lógicas de segurança mais complexas (ex: alarme que exige uma sequência para desativar).
 - **5. Cultura Digital:** Aprofundar a compreensão da interface humana-computador e como hardware simples pode emular inputs complexos. Projetar interfaces não-convencionais.

- **6. Trabalho e Projeto de Vida:** Planejar e executar um sistema de segurança simples, considerando seus pontos fortes e fracos.
 - **7. Argumentação:** Justificar a escolha de materiais e a lógica de programação para um sistema de alarme mais robusto.
- **Competências Específicas de Matemática e suas Tecnologias:**
 - **Habilidades Relacionadas a Álgebra:**
 - Modelar sistemas lógicos complexos (circuitos de segurança com múltiplos pontos de entrada), utilizando proposições lógicas e operadores booleanos (AND, OR, NOT).
 - Analisar a lógica de ativação e desativação de um alarme.
 - **Habilidades Relacionadas a Funções:**
 - Compreensão de funções de entrada/saída discreta (um toque dispara uma ação).
 - **Habilidades Relacionadas à Probabilidade e Estatística:**
 - Discussão sobre a probabilidade de falha ou falso alarme em sistemas simples e como a lógica pode mitigar isso.

Materiais Necessários:

- 1 Placa Makey Makey
- 1 Cabo USB para Makey Makey
- 2 Cabos jacaré
- 1 Computador com Pictoblox instalado e com internet
- 1 Caixa de papelão
- Papel Alumínio (ou fita adesiva condutora)

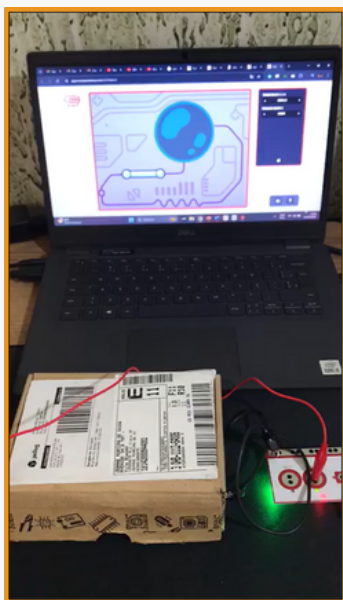
Passo a Passo da Montagem do Circuito:

1. Conecte a Makey Makey: Conecte a Makey Makey ao seu computador via cabo USB. O computador o reconhecerá como um teclado/mouse padrão.

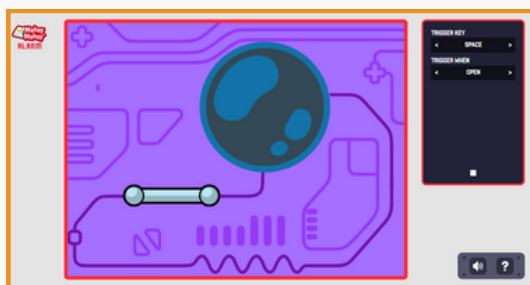
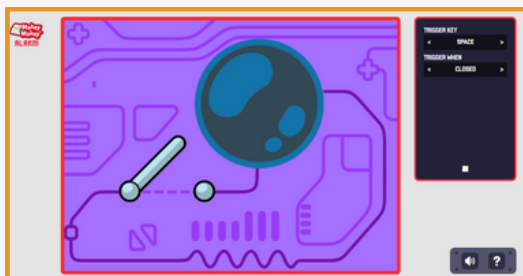
2. Conecte o "terra": Use um cabo jacaré para conectar o pino terra da Makey Makey a algo que você ou o usuário irá segurar constantemente. Isso pode ser uma pulseira ajustada ao corpo, um anel, um pedaço de massinha que a pessoa segure, ou a própria mão (segurando a ponta do cabo). Este é o ponto de referência do circuito.

3. Conecte os Sensores:

- Monte a caixa de papelão conforme indicando na imagem, observando os pontos de contato em bordas específicas da caixa.
- O objetivo é de que, quando essas bordas não estiverem se tocando (simulando a caixa aberta), o alarme seja disparado.
- Conecte uma das pontas do cabo jacaré a somente uma dessas bordas e a outra ponta em uma das entradas de "tecla" no Makey Makey (espaço, clique do mouse ou setas).



Para esta simulação, utilizamos a princípio, o Sistema de Alarme da base de aplicativos Plug & Play do site da Makey Makey. É possível acessá-lo (igualmente a inúmeros outros exemplos), pelo link <https://apps.makeymakey.com/v2/#alarm>.



Utilizando uma caixa de papelão virtual, isto é, com figuras baixadas de sites de figuras gratuitas de sua preferência, nos modos aberta e fechada, vamos simular diretamente no Pictoblox a programação abaixo. A essência do problema é a mesma mas, neste caso, não teremos uma caixa “física” mas, virtual.

Também usamos imagens de sirene, texto (“caixa aberta” e “caixa fechada”) e de cabo jacaré, cada um com sua devida programação, para simularmos o alarme disparado de maneira sonora e visual.

Passo a Passo da Programação no Pictoblox:

- 1. Abra o Pictoblox:** Inicie o software Pictoblox em seu computador.
- 2. Selecione o Dispositivo Makey Makey:** No menu "Dispositivos", selecione "Makey Makey".
- 3. Adicionar Extensões:** Certifique-se de ter a extensão "Makey Makey" ativada.
- 4. Segue abaixo a programação de cada ator:** caixa, sirene, placa Makey Makey, textos e cabos jacaré.

```
quando a tecla espaço for pressionada
se tecla espaço pressionada? então
  muda para a fantasia sirene25
senão
  próxima fantasia
```

```
quando a tecla espaço for pressionada
se tecla espaço pressionada? então
  muda para a fantasia jacare2
senão
  muda para a fantasia jacare1
```

```
quando a tecla espaço for pressionada
se tecla espaço pressionada? então
  muda para a fantasia makey_1
senão
  muda para a fantasia makey_2
```

```
quando a tecla espaço for pressionada
se tecla espaço pressionada? então
  muda para a fantasia alerta_2
senão
  muda para a fantasia alerta_1
```

```
quando a tecla espaço for pressionada
se tecla espaço pressionada? então
  muda para a fantasia caixa 1
senão
  muda para a fantasia caixa 2
  toque o som Alert até o fim
```



Observe que, enquanto estivermos tocando a tecla espaço, na Makey Makey física, a Makey Makey virtual não mudará de fantasia, igualmente os demais atores, simulando a caixa de papelão virtual fechada, isto é, não emitirá som ou luz pela sirene. Quando deixarmos de tocar a tecla espaço, os atores mudarão de fantasia imediatamente.



Testando o Projeto:

Com a Makey Makey conectada e o código carregado no Pictoblox:

- Segure o cabo jacaré conectado ao pino "terra" do Makey Makey.
- Toque no objeto condutor que você conectou ao pino da tecla (por exemplo, a tecla espaço).
- Você deverá ouvir o som do alarme e/ou ver a mensagem ou ação programada no Palco.

Experimente tocar o objeto condutor sem segurar o "terra", para demonstrar que o circuito não é fechado e o alarme não dispara.

Sugestões de Aplicação em Sala de Aula de Matemática:

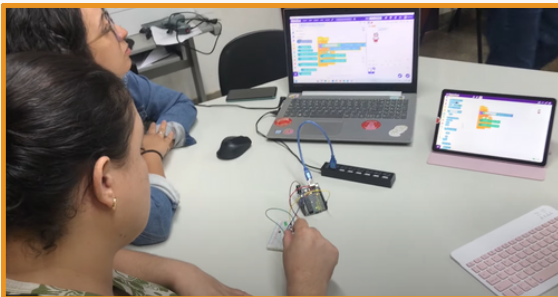
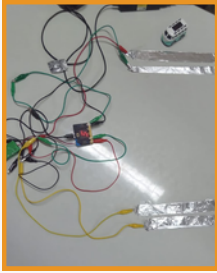
- **Conceitos de Circuito e Condutividade:** Explorar quais materiais são condutores (ligam o alarme) e quais são isolantes (não ligam), e como isso se relaciona com os números e as propriedades dos materiais.
- **Lógica "Se-Então":** Desenvolver cenários matemáticos que exigem uma condição para acontecer (ex: se o número for par, então toque o alarme).
- **Sistemas de Senhas Numéricas:** Criar um "teclado" de papel com números em grafite. O alarme só desativa se uma sequência numérica correta for tocada (requer mais blocos se...então e variáveis para armazenar a sequência digitada).
- **Contadores de Eventos:** Fazer com que o alarme conte quantas vezes foi acionado, registrando o número de "intrusões" e exibindo-o. Isso leva a noções de coleta de dados e estatística básica.
- **Jogos de Correspondência:** Criar um jogo em que o aluno precisa tocar no "sensor" correto que corresponde a uma resposta matemática (ex: "Qual a raiz quadrada de 25?" - tocar no sensor "5").

Desafios/Variações:

- **Alarme Silencioso:** Em vez de um som, o alarme pode apenas enviar uma mensagem para um "monitor" no Pictoblox ou mudar um sprite escondido.
- **Alarme com Temporizador:** Fazer com que o alarme toque por um tempo limitado e depois pare automaticamente, a menos que seja reativado.
- **Alarme Multiponto:** Conectar vários objetos condutores a diferentes pinos do Makey Makey para criar um sistema de alarme com múltiplas "zonas" (ex: pino da seta para cima para a porta, pino da seta para baixo para a janela).
- **Jogo da Memória:** Criar um jogo da memória onde tocar pares de objetos condutores (com o mesmo conceito matemático, por exemplo) faz com que eles se revelem no palco.
- **Instrumento Musical Matemático:** Criar um piano ou bateria com objetos condutores, onde cada "tecla" toca uma nota ou som e, ao mesmo tempo, exibe uma equação ou resultado matemático.

Alguns registros do 2º. Encontro:





ENCONTRO 3: SENSORES E DADOS PARA UM AMBIENTE SUSTENTÁVEL

O terceiro encontro do curso aprofundou a aplicação da robótica educacional para a compreensão e intervenção em questões ambientais, com o tema "Sensores e Dados para um Ambiente Sustentável". Os projetos focaram na coleta e análise de dados para monitorar condições e promover a conscientização coletiva.

Projeto 3.1: Monitoramento de Temperatura e Umidade

Placa Utilizada: Arduino Uno

Objetivo de Aprendizagem:

- **Matemático:** Compreender e analisar grandezas físicas (temperatura, umidade, concentração de gases), realizar coleta e organização de dados, construir gráficos, e explorar conceitos de média, moda, mediana e desvio padrão.
- **Computacional:** Desenvolver habilidades de programação em blocos (Pictoblox) para leitura de múltiplos sensores analógicos, exibição de dados (via Monitor Serial ou display), e utilização de lógicas para acionar alertas com base em limites de segurança.

Conceitos Matemáticos Envolvidos:

- **Grandezas e Medidas:** Temperatura (graus Celsius) e Umidade (porcentagem).
- **Coleta e Organização de Dados:** Registros contínuos ou em intervalos definidos.
- **Estatística Descritiva:** Cálculo de média, identificação de valores máximos/mínimos, análise de variação (amplitude), e introdução a desvio padrão (para alunos mais avançados).

- **Gráficos:** Representação visual de dados ao longo do tempo (gráficos de linha, de barras) para identificar tendências e padrões.
- **Funções:** A relação entre o valor lido pelo sensor e a grandeza física que ele representa.
- **Inequações/Limites:** Definição de zonas de conforto ou de alerta para as variáveis ambientais.

Competências e Habilidades da BNCC:

Este projeto, utilizando o Arduino Uno com múltiplos sensores, é ideal para o Ensino Médio e pode ser simplificado para o Ensino Fundamental com foco em medição e registro de dados básicos.

Para o Ensino Médio:

- **Competências Gerais:**
 - **2. Pensamento Científico, Crítico e Criativo:** Investigar problemas ambientais reais, coletar e analisar dados de forma sistemática para tirar conclusões e propor soluções baseadas em evidências.
 - **5. Cultura Digital:** Utilizar o hardware (Arduino e sensores) e o software (Pictoblox) para coletar, processar e visualizar grandes volumes de dados, compreendendo o papel da tecnologia na monitorização ambiental.
 - **7. Argumentação:** Debater sobre a qualidade do ar, a temperatura ideal e a umidade, usando os dados coletados como base para argumentos e proposições.
 - **10. Responsabilidade e Cidadania:** Conscientizar sobre questões ambientais urgentes, como poluição do ar e conforto térmico, e propor ações para um ambiente mais sustentável.

- **Competências Específicas de Matemática e suas Tecnologias:**

- **Habilidades Relacionadas a Grandezas e Medidas:**

- Utilizar diferentes unidades de medida (temperatura, umidade) e compreender a escala de valores para representar as grandezas.
- Resolver problemas envolvendo conversões de unidades, se necessário.

- **Habilidades Relacionadas à Probabilidade e Estatística:**

- Interpretar conjuntos de dados bivariados (temperatura vs. tempo, umidade vs. tempo), identificar tendências, dispersão e anomalias.
- Calcular e interpretar medidas de tendência central (média, mediana, moda) e dispersão (amplitude, desvio padrão) dos dados coletados.
- Construir e interpretar diferentes tipos de gráficos (linha, barras) para representar a evolução das variáveis ambientais.

- **Habilidades Relacionadas a Álgebra e Funções:**

- Modelar a relação entre as variáveis ambientais e o tempo (funções de tempo), e analisar como essas funções se comportam.
- Aplicar conceitos de limites e inequações para definir faixas de valores seguras ou de alerta para temperatura, umidade e gases.

Para o Ensino Fundamental:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Observar e registrar variações de temperatura e umidade, formulando hipóteses simples sobre o que causa essas mudanças.
- **5. Cultura Digital:** Primeiros passos na coleta de dados com sensores e a visualização desses dados no computador.

- **10. Responsabilidade e Cidadania:** Discutir a importância de manter um ambiente saudável e seguro em casa e na escola.
- **Competências Específicas de Matemática:**
 - **Grandezas e Medidas:** Leitura e interpretação de valores de temperatura e umidade. Comparação de "mais quente/mais frio", "mais úmido/menos úmido".
 - **Números:** Registro e organização de dados numéricos em tabelas.
 - **Probabilidade e Estatística:** Construção de gráficos de barras simples para representar as variações diárias ou semanais, identificando o valor mais frequente.
 - **Álgebra:** Introdução à ideia de se/então para acionar um alerta simples (ex: se temperatura > 30, então acender LED).

Materiais Necessários:

- 1 Arduino Uno
- 1 Protoboard (400 pontos)
- 2 LEDs (5mm - 20mA - 5V)
- 2 Resistores 220Ω (ohms)
- 6 Jumpers (Macho x Macho)
- 1 Sensor de temperatura e umidade (DHT11)

Passo a Passo da Montagem do Circuito:

1. Montagem do Sensor DHT11 (Temperatura e Umidade):

- O sensor DHT11 possui 4 pinos: VCC, GND, DATA e NC.
- Conecte o pino VCC do DHT ao 5V do Arduino Uno.
- Conecte o pino GND do DHT ao GND do Arduino Uno.
- Conecte o pino Data do DHT a uma porta analógica do Arduino Uno (pino A0).

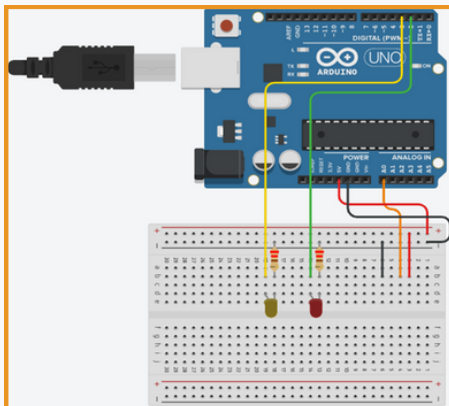
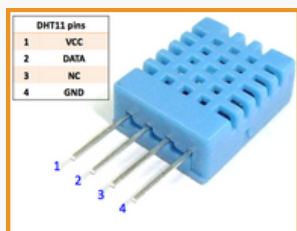
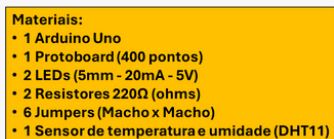
2. Montagem dos LEDs de Alerta:

- Conecte cada perna mais longa (ânodo) dos LEDs às portas digitais 2 e 3, respectivamente, do Arduino Uno.

- Conecte a perna mais curta (cátodo) de cada LED a um resistor de 220 Ohm.
- Conecte a outra ponta de cada resistor ao GND do Arduino Uno.

3. Conexão Final:

- Verifique todas as conexões.
- Conecte o cabo USB do Arduino Uno ao seu computador.



Passo a Passo da Programação no Pictoblox:

1. Abra o Pictoblox: Inicie o software Pictoblox em seu computador.

2. Selecione o Dispositivo Arduino: No menu "Dispositivos", selecione "Arduino" e, em seguida, "Arduino Uno" (ou o modelo que estiver usando). Conecte-o via USB ao computador.

3. Adicionar Extensões: Para interagir com o Arduino Uno, você precisará adicionar as extensões de "Arduino" e, opcionalmente, de "Display" para visualizar valores.

4. Estrutura Básica:

- Comece criando dois blocos de evento quando o Arduino Uno iniciar.
- Dentro destes, use um bloco sempre para que o programa monitore continuamente o sensor DHT11.
- Crie as variáveis "temperatura" e "humidade", para serem usadas em cada conjunto de blocos de iteração.

- Use o bloco diga, combinados com os blocos junte, para que o ator diga os valores de temperatura e humidade, registrados à partir do pino analógico A0.
- Em cada conjunto de blocos, use o bloco condicional se/então/senão, com os limiares definidos abaixo.

5. Leitura do Sensor DHT11:

- Para o primeiro conjunto de condicionais, estabeleça que, se o valor da temperatura for maior ou igual a 30°C, então o pino digital 2, isto é, o LED (vermelho) vai acender. Caso contrário, permanecerá apagado.
- Para o segundo conjunto de condicionais, estabeleça que, se o valor da humidade do ar for maior ou igual a 50%, então o pino digital 3, isto é, o LED (amarelo) vai acender. Caso contrário, permanecerá apagado.

```

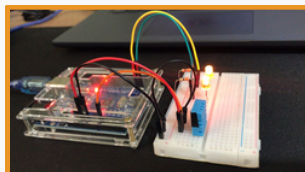
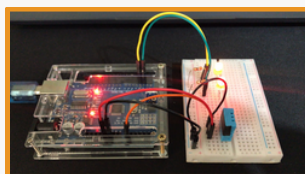
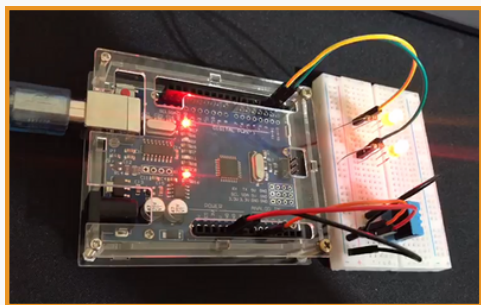
quando for clicado
  sempre
    mude temperatura para obter temperatura de sensor DHT no pino A0
    diga junta junta Temperatura com obter temperatura de sensor DHT no pino A0 com 1
    se temperatura > 30 ou temperatura = 30 então
      Definição de pino digital 2 saída como ALTO
    senão
      Definição de pino digital 2 saída como BAIXO
  
```

```

quando for clicado
  sempre
    mude umidade para obter umidade de sensor DHT no pino A0
    diga junta junta Humidade com obter umidade de sensor DHT no pino A0 com %
    se umidade > 50 ou umidade = 50 então
      Definição de pino digital 3 saída como ALTO
    senão
      Definição de pino digital 3 saída como BAIXO
  
```



Na Placa:



Testando o Projeto:

Após carregar o código para o Arduino Uno:

- O sistema começará a ler os valores de temperatura, umidade pelo sensor DHT11.
- Ative cada bloco de interações separadamente, para que não haja conflito na exibição (pelo ator do Pictoblox) do valor analógico lido, que serão exibidos no Palco do Pictoblox.
- Se a temperatura ou a umidade excederem os limites definidos no código, os LEDs de alerta acenderão. Caso contrário, permanecerão apagados.
- É importante colocar um bloco de aguarde 1 segundo quando o LED for acionado, isto é, na condição se/então verdadeira, para que ele se apague e aguarde a nova leitura.
- Experimente variar as condições ambientais (soprar ar quente e úmido no sensor) para ver as leituras mudarem e os LEDs variarem entre ligado e desligado.

Sugestões de Aplicação em Sala de Aula de Matemática:

- **Coleta e Organização de Dados:** Criar tabelas para registrar as leituras de temperatura, umidade e gás ao longo do tempo (ex: a cada hora por um dia).

- **Construção e Análise de Gráficos:** Transformar os dados coletados em gráficos de linha para visualizar tendências diárias ou semanais. Comparar a temperatura interna e externa da sala de aula.
- **Medidas de Tendência Central e Dispersão:** Calcular a média, moda e mediana das temperaturas registradas. Discutir a amplitude térmica do ambiente. Para alunos mais avançados, introduzir o conceito de desvio padrão.
- **Problemas de Otimização e Conforto Térmico:** Discutir a temperatura e umidade ideais para um ambiente. Analisar se o sistema de monitoramento ajuda a manter o conforto e como otimizar o uso de ar condicionado/aquecimento com base nos dados.
- **Análise de Dados Ambientais Reais:** Conectar o projeto a discussões sobre a qualidade de vida em sua cidade, os efeitos das mudanças climáticas, usando os dados como base para a compreensão.
- **Inequações Aplicadas:** Trabalhar com os "limites" que acionam o alarme, explorando o conceito de inequações (ex: Temperatura $> 30^{\circ}\text{C}$).

Desafios/Variações:

- Integrar um display LCD para mostrar as leituras de temperatura e umidade diretamente no dispositivo, sem a necessidade do computador.
- Adicionar um pequeno ventilador controlado pelo Arduino que liga automaticamente se a temperatura ultrapassar um certo limite.
- Desenvolver um "diário climático" onde o Arduino registra os dados em um cartão SD e os alunos podem depois analisar esses dados no computador.
- Conectar o Arduino a uma plataforma de IoT para enviar os dados para a nuvem e visualizá-los em dashboards online, explorando a matemática em Big Data.

Projeto 3.2: Monitoramento de Gases Inflamáveis

Placa Utilizada: Arduino Uno

Objetivo de Aprendizagem:

- **Matemático:** Compreender e analisar grandezas físicas (concentração de gases), realizar coleta e organização de dados, construir gráficos, e explorar conceitos de média, moda, mediana e desvio padrão.
- **Computacional:** Desenvolver habilidades de programação em blocos (Pictoblox) para leitura de múltiplos sensores analógicos, exibição de dados (via Monitor Serial ou display), e utilização de lógicas para acionar alertas com base em limites de segurança.

Conceitos Matemáticos Envolvidos:

- **Grandezas e Medidas:** Concentração de Gases (valores analógicos).
- **Coleta e Organização de Dados:** Registros contínuos ou em intervalos definidos.
- **Estatística Descritiva:** Cálculo de média, identificação de valores máximos/mínimos, análise de variação (amplitude), e introdução a desvio padrão (para alunos mais avançados).
- **Gráficos:** Representação visual de dados ao longo do tempo (gráficos de linha, de barras) para identificar tendências e padrões.
- **Funções:** A relação entre o valor lido pelo sensor e a grandeza física que ele representa.
- **Inequações/Limites:** Definição de zonas de conforto ou de alerta para as variáveis ambientais.

Competências e Habilidades da BNCC:

Este projeto, utilizando o Arduino Uno com múltiplos sensores, é ideal para o Ensino Médio pois foca na medição e registro de dados básicos.

Para o Ensino Médio:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Investigar problemas ambientais reais, coletar e analisar dados de forma sistemática para tirar conclusões e propor soluções baseadas em evidências.
- **5. Cultura Digital:** Utilizar o hardware (Arduino e sensores) e o software (Pictoblox) para coletar, processar e visualizar grandes volumes de dados, compreendendo o papel da tecnologia na monitorização ambiental.
- **7. Argumentação:** Debater sobre a qualidade do ar, a temperatura ideal e a umidade, usando os dados coletados como base para argumentos e proposições.
- **10. Responsabilidade e Cidadania:** Conscientizar sobre questões ambientais urgentes, como poluição do ar e conforto térmico, e propor ações para um ambiente mais sustentável.

- **Competências Específicas de Matemática e suas Tecnologias:**

- **Habilidades Relacionadas a Grandezas e Medidas:**
 - Utilizar diferentes unidades de medida (temperatura, umidade) e compreender a escala de valores para representar as grandezas.
 - Resolver problemas envolvendo conversões de unidades, se necessário.
- **Habilidades Relacionadas à Probabilidade e Estatística:**
 - Interpretar conjuntos de dados bivariados (temperatura vs. tempo, umidade vs. tempo), identificar tendências, dispersão e anomalias.
 - Calcular e interpretar medidas de tendência central (média, mediana, moda) e dispersão (amplitude, desvio padrão) dos dados coletados.

- Construir e interpretar diferentes tipos de gráficos (linha, barras) para representar a evolução das variáveis ambientais.
- **Habilidades Relacionadas a Álgebra e Funções:**
 - Modelar a relação entre as variáveis ambientais e o tempo (funções de tempo), e analisar como essas funções se comportam.
 - Aplicar conceitos de limites e inequações para definir faixas de valores seguras ou de alerta para temperatura, umidade e gases.

Materiais Necessários:

- 1 Arduino Uno
- 1 Protoboard (400 pontos)
- 1 LED (5mm - 20mA - 5V)
- 1 Resistor 220Ω (ohms)
- 8 Jumpers (Macho x Macho)
- 1 Sensor de Gás Inflamável (MQ2)
- 1 Sensor de Som (Buzzer Ativo)

Passo a Passo da Montagem do Circuito:

1. Montagem do Sensor MQ2 (Sensor de Gás Inflamável):

- O sensor MQ2 possui 4 pinos: VCC, GND, D0 e A0.
- Conecte o pino VCC do MQ2 ao 5V do Arduino Uno.
- Conecte o pino GND do MQ2 ao GND do Arduino Uno.
- Conecte o pino A0 do MQ2 a uma porta analógica do Arduino Uno (pino A0).

2. Montagem do Buzzer (Sensor de Som):

- Conecte cada perna mais longa (ânodo +) do Buzzer à porta digital 8 do Arduino Uno.
- Conecte cada perna mais curta (cátodo -) do Buzzer ao GND do Arduino Uno.

3. Montagem do LED:

- Conecte a perna mais longa (ânodo) do LED à porta digital 7 do Arduino Uno.
- Conecte a perna mais curta (cátodo) do LED à um resistor de 220 Ohm.

- Conecte a outra ponta do resistor ao GND do Arduino.

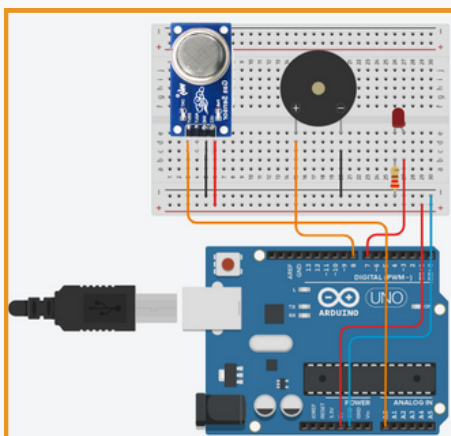
4. Conexão Final:

- Verifique todas as conexões.
- Conecte o cabo USB do Arduino Uno ao seu computador.



Materiais:

- 1 Arduino Uno
- 1 Protoboard (400 pontos)
- 1 LED (5mm - 20mA - 5V)
- 1 Resistor 220Ω (ohms)
- 8 Jumpers (Macho x Macho)
- 1 Sensor de Gás (MQ2)
- 1 Sensor de Som (Buzzer)



Passo a Passo da Programação no Pictoblox:

1. Abra o Pictoblox: Inicie o software Pictoblox em seu computador.

2. Selecione o Dispositivo Arduino: No menu "Dispositivos", selecione "Arduino" e, em seguida, "Arduino Uno" (ou o modelo que estiver usando). Conecte-o via USB ao computador.

3. Adicionar Extensões: Para interagir com o Arduino Uno, você precisará adicionar as extensões de "Arduino" e, opcionalmente, de "Display" para visualizar valores.

4. Estrutura Básica:

- Comece criando um bloco de evento quando o Arduino Uno iniciar.
- Use um bloco sempre para que o programa monitore continuamente o sensor MQ2.
- Crie a variável "gás", para ser usada no bloco de iteração.
- Estabeleça que a variável "gás" receba a leitura do sensor de MQ2 pelo pino analógico A0.

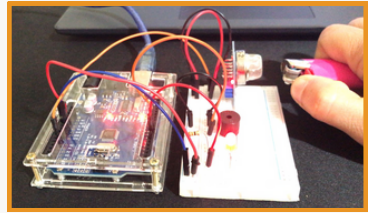
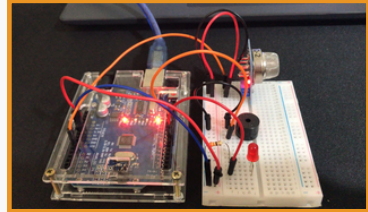
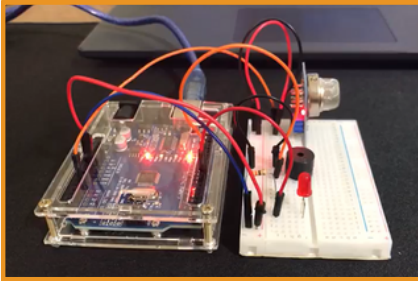
5. Leitura do Sensor MQ2:

- Para o conjunto de condicionais se/então, estabeleça que, se o sensor MQ2 registrar uma concentração de gases inflamáveis, como metano, butano, propano, etc., ou fumaça, maior que 65, então os pinos digitais 7 e 8, serão definidos como alto. Isto é, o LED (vermelho) vai acender e o Buzzer vai emitir um som de alerta.
- Isso vai acontecer em intervalos de 1 segundo, até que não seja mais detectado qualquer sinal de gás inflamável.
- Caso contrário, o LED permanecerá apagado e o Buzzer sem emitir som, devido ao uso dos blocos de definição dos pinos 7 e 8 como baixo.

OBS: para esse experimento, será necessário o uso de um isqueiro (ou fósforo ou vela), para que sejam produzidos elementos inflamáveis para o sensor detectar. Por esse motivo, este experimento não é recomendado para alunos do ensino fundamental (ou crianças). Mesmo com alunos do ensino médio, deve ser realizado sempre com a supervisão de um professor.

The image displays a Scratch script for an MQ2 gas sensor and a corresponding user interface. The script is enclosed in a yellow box and begins with a 'quando for clicado' (when clicked) event. It then enters a 'sempre' (forever) loop. Inside the loop, the first block is 'mude gas para ler sensor analógico sensor de gás no pino A0', which sets the 'gas' variable to the value of the 'sensor de gás' sensor on pin A0. This is followed by a conditional 'se gas > 65 então' (if gas > 65 then). If true, it triggers a 'diga Gás Inflamável detectado!' (say 'Gás Inflamável detectado!') block. Next, two 'Definir pino digital' (set digital pin) blocks are used to set pins 7 and 8 to 'saída como ALTO' (output as HIGH). This is followed by an 'espere 1 seg' (wait 1 second) block. Then, two more 'Definir pino digital' blocks set pins 7 and 8 to 'saída como BAIXO' (output as LOW). Another 'espere 1 seg' block follows. Finally, a 'senão' (else) block sets pins 7 and 8 to 'saída como BAIXO'. The UI mockup, shown in a white box with an orange border, features a 'gas' variable monitor displaying the value '78'. Below it is a cartoon bear wearing goggles and a red shirt, with the text 'Gás Inflamável detectado!' (Inflammable gas detected!).

Na Placa:



Testando o Projeto:

Após carregar o código para o Arduino Uno:

- O sistema começará a tentar detectar valores de gás inflamável ou fumaça.
- Se for detectado e exceder o limite definido no código, o LED de alerta acenderá e o buzzer emitirá um som de alerta. Caso contrário, eles permanecerão desligados.
- Neste caso, para segurança, é sempre importante deixar o limiar de detecção de gás o mais baixo possível.
- É importante colocar um bloco de aguarda 1 segundo quando o LED for acionado, ou seja, na condição se/então verdadeira, para que ele se apague e acenda continuamente, enquanto houver leitura por parte do sensor.
- Experimente variar as condições ambientais (soprar ar quente e úmido no sensor) para ver as leituras mudarem e os LEDs variarem entre ligado e desligado.

Sugestões de Aplicação em Sala de Aula de Matemática:

- **Coleta e Organização de Dados:** Criar tabelas para registrar as leituras da ocorrência de gases inflamáveis pelo ambiente ao longo do tempo (ex: a cada hora por um dia).

- **Construção e Análise de Gráficos:** Transformar os dados coletados em gráficos de linha para visualizar tendências diárias ou semanais. Discutir quando esses eventos ocorreram e os motivos para essa ocorrência e, se foram externas ou internas à sala de aula.
- **Análise de Dados Ambientais Reais:** Conectar o projeto a discussões sobre a qualidade de vida em sua cidade, os efeitos das mudanças climáticas, usando os dados como base para a compreensão.
- **Inequações Aplicadas:** Trabalhar com os "limites" que acionam o alarme, explorando o conceito de inequações.

Desafios/Variações:

- Integrar um display LCD para mostrar as leituras de temperatura e umidade diretamente no dispositivo, sem a necessidade do computador.
- Criar um sistema de alerta para diferentes níveis de concentração de gás (ex: LED amarelo para alerta, LED vermelho para perigo).
- Desenvolver um "diário climático" onde o Arduino registra os dados em um cartão SD e os alunos podem depois analisar esses dados no computador.
- Conectar o Arduino a uma plataforma de IoT para enviar os dados para a nuvem e visualizá-los em dashboards online, explorando a matemática em Big Data.

Projeto 3.3: Sensor de Luz e Termômetro

Placa Utilizada: Micro:bit

Objetivo de Aprendizagem:

- **Matemático:** Coletar, organizar e interpretar dados quantitativos (temperatura, níveis de luz), entender a importância da amostragem e do registro sistemático, e visualizar dados para identificar padrões e tendências.
- **Computacional:** Utilizar os sensores internos da Micro:bit para obter leituras, manipular variáveis para armazenar e exibir dados, e compreender o conceito de registro e transmissão de informações.

Conceitos Matemáticos Envolvidos:

- **Grandezas e Medidas:** Temperatura (graus Celsius), Nível de Luz (valores analógicos).
- **Coleta de Dados:** Amostragem de dados em intervalos regulares.
- **Organização de Dados:** Criação de tabelas e listas para registrar as leituras.
- **Estatística Descritiva:** Identificação de valores mínimos, máximos e cálculo de média para análise dos dados.
- **Gráficos Simples:** Possibilidade de esboçar gráficos de linha ou de barras a partir dos dados coletados.
- **Intervalos e Frequência:** Registrar dados a cada intervalo de tempo estabelecido (segundos/minutos/hora/dia).

Competências e Habilidades da BNCC:

Este projeto, com a Micro:bit, é altamente relevante para o Ensino Fundamental e o Ensino Médio, com aprofundamento na análise estatística para os anos finais e Ensino Médio.

Para o Ensino Fundamental:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Observar e registrar fenômenos naturais (temperatura, luz), formular perguntas sobre as variações e buscar padrões nos dados.
- **4. Comunicação:** Apresentar os dados coletados de forma clara, utilizando a matriz de LEDs ou o Monitor Serial do Pictoblox.
- **5. Cultura Digital:** Utilizar a Micro:bit como ferramenta para coletar dados do ambiente e interagir com eles de forma programada.
- **10. Responsabilidade e Cidadania:** Discutir a importância de monitorar o ambiente e como pequenas ações podem contribuir para a sustentabilidade.

- **Competências Específicas de Matemática:**

- **Grandezas e Medidas:** Leitura de valores de temperatura e luz, e compreensão de como representam a intensidade dessas grandezas.
- **Números:** Registro e organização de sequências numéricas em tabelas simples. Comparação de valores.
- **Probabilidade e Estatística:** Coleta de dados sobre o ambiente, identificação de valores mais frequentes (moda) e discussão sobre variações diárias. Construção de gráficos de barras simples.
- **Álgebra:** Uso de variáveis para armazenar as leituras dos sensores.

Para o Ensino Médio:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Desenvolver experimentos de coleta de dados ambientais, analisar criticamente as informações e anali-

- analisar criticamente as informações e inferir tendências ou anomalias.
 - **5. Cultura Digital:** Utilizar a Micro:bit como um dispositivo de coleta de dados para estudos ambientais, compreendendo o processo de aquisição e transmissão de dados.
 - **7. Argumentação:** Debater sobre a qualidade dos dados, a metodologia de coleta e as conclusões tiradas a partir das observações.
 - **10. Responsabilidade e Cidadania:** Aprofundar a discussão sobre monitoramento ambiental, mudanças climáticas, poluição e o papel da tecnologia na conscientização e mitigação de problemas.
- **Competências Específicas de Matemática e suas Tecnologias:**
 - **Habilidades Relacionadas à Probabilidade e Estatística:**
 - Planejar e executar experimentos de coleta de dados, organizando-os em tabelas e gráficos (linha, dispersão) para análise.
 - Calcular e interpretar medidas de tendência central e de dispersão dos dados coletados, inferindo conclusões sobre os fenômenos ambientais.
 - Analisar a correlação entre diferentes variáveis (ex: temperatura e nível de luz ao longo do dia).
 - **Habilidades Relacionadas a Funções:**
 - Modelar a variação de temperatura ou luz ao longo do tempo como uma função, compreendendo o comportamento e as características (periodicidade, amplitude).
 - **Habilidades Relacionadas a Grandezas e Medidas:**
 - Compreender a precisão dos sensores e as possíveis fontes de erro nas medições, discutindo a propagação de erros.

Materiais Necessários:

- 1 BBC Micro:bit
- 2 Pilhas AAA
- Suporte para Baterias AAA

Passo a Passo da Montagem do Circuito:

Assim como em projetos anteriores com uso da Micro:bit, este projeto se beneficia dos sensores integrados da placa. A montagem é mínima, isto é, basta conectar a Micro:bit ao computador via cabo USB.

Passo a Passo da Programação no Pictoblox:

1. Abra o Pictoblox: Inicie o software Pictoblox em seu computador.

2. Selecione o Dispositivo Micro:bit: No menu "Dispositivos", selecione "Micro:bit". Conecte-o via USB ao computador.

3. Adicionar Extensões: Certifique-se de ter a extensão "Micro:bit" ativada.

4. Estrutura Básica:

- Comece com o bloco de evento quando a Micro:bit iniciar.
- Use um bloco sempre para leitura contínua dos sensores.

5. Exibição dos Dados na Matriz de LEDs (opcional, para visualização rápida):

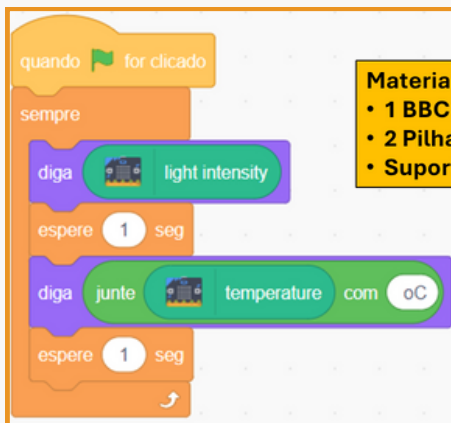
- A Micro:bit possui blocos diretos para leitura de temperatura e nível de luz.
- Combine dois blocos: "diga" e "light intensity" para apresentar o nível de luz do ambiente na matriz de LEDs da Micro:bit.
- Use o bloco "esperar 1 segundo" para mostrar por um tempo o valor da luminosidade do ambiente.
- Combine dois blocos: "diga", "junte (temperature) com (°C)" para exibir o valor da temperatura do ambiente, na matriz de LEDs da Micro:bit.

- Use o bloco “esperar 1 segundo” para mostrar por um tempo o valor da temperatura do ambiente.

6. Leitura dos Sensores Internos via botões de programação:

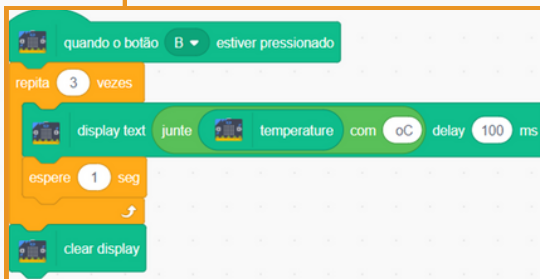
- Também é possível fazer a exibição dos valores lidos pelos sensores, via acionamento dos botões A e B da Micro:bit, condicionados à limiares estabelecidos, com resposta na matriz de LEDs da Micro:bit.
- Use como bloco principal “quando o botão A estiver pressionado”.
- Use o bloco sempre para leitura contínua do sensor de luminosidade.
- Em seguida, estabeleça a condição lógica para exibição da luminosidade, combinando os blocos: “se light intensity > (60) então”.
- Na condição acima verdadeira, use o bloco “repita por (3) vezes” para leitura e exibição do valor da luminosidade do ambiente.
- Dentro do repita, use os blocos: “exiba ícone” da sua preferência, para representar a intensa luminosidade.
- Use o bloco “espere 1 segundo”, como tempo de exibição do ícone.
- Se preferir, também exiba o ícone anterior, em dimensões menores, simbolizando um “brilho” da luminosidade.
- Fora do repita, use o bloco “clear display” para limpar a tela, que ficará sem exibir nenhum ícone, até novo acionamento do botão A.
- Para a temperatura, use como bloco principal “quando o botão B estiver pressionado”.
- Use o bloco “repita (3) vezes” para leitura e exibição do valor da temperatura do ambiente.
- Dentro do repita, combine os blocos: “display text”, “junte (temperature) com (°C)” para exibir a temperatura na matriz de LEDs da Micro:bit.
- Use o bloco “espere 1 segundo”, como tempo de exibição do valor da temperatura.

- o Fora do repita, use o bloco “clear display” para limpar a tela, que ficará sem exibir nenhum ícone, até novo acionamento do botão A.

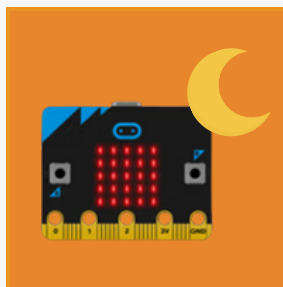
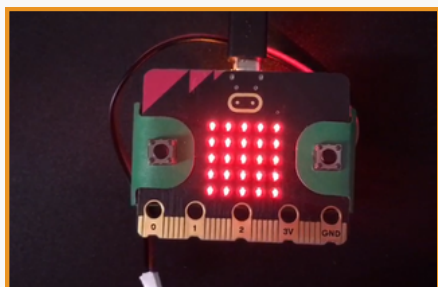


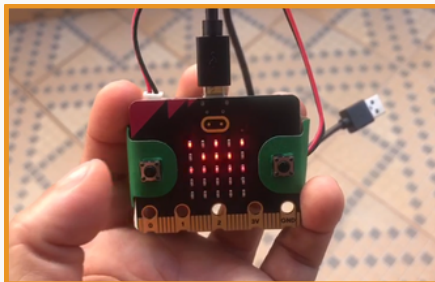
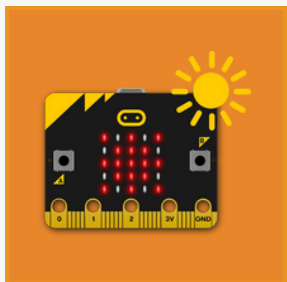
Materiais:

- 1 BBC Micro:bit
- 2 Pilhas AAA
- Suporte para Baterias AAA



Na Placa:





Testando o Projeto:

Com a Micro:bit conectada e o código carregado:

- A Micro:bit começará a exibir a temperatura e o nível de luz na matriz de LEDs.
- No Pictoblox, abra o "Monitor Serial" (geralmente em "Conectar Dispositivo" ou "Extensões"). Você verá os dados sendo registrados em tempo real.
- Experimente variar a temperatura ambiente (soprar na placa) ou o nível de luz (cobrir o sensor de luz na parte frontal da placa) e observe as leituras mudarem.

Sugestões de Aplicação em Sala de Aula de Matemática:

- **Coleta e Análise de Dados Reais:** Realizar um estudo ambiental na própria sala de aula: onde é mais quente/frio? Mais claro/escuro? Como a temperatura e a luz variam ao longo do dia? Registrar esses dados sistematicamente.
- **Construção de Gráficos:** Utilizar os dados coletados para criar gráficos de linha (temperatura ou luz versus tempo) ou gráficos de dispersão, explorando a relação entre as variáveis.
- **Cálculo de Médias e Medidas de Dispersão:** Calcular a temperatura média da sala, a variação (amplitude) e, para alunos do Ensino Médio, o desvio padrão.
- **Comparação de Ambientes:** Comparar os dados coletados em diferentes partes da escola (sala de aula, corredor, pátio) para analisar as variações ambientais.

- **Problemas de Modelagem:** Desafiar os alunos a criar um modelo que preveja a temperatura em um determinado horário do dia com base nos dados históricos.
- **Interpretação de Tendências:** Identificar se há tendências de aumento ou diminuição de temperatura/luz ao longo de um período.
- **Discussão de Amostragem:** Por que é importante registrar os dados em intervalos regulares? O que aconteceria se registrássemos a cada 1 segundo? E a cada 1 hora?

Desafios/Variações:

- Programar a Micro:bit para exibir um ícone de "calor" ou "frio" excessivo, ou emitir um som de alerta se a temperatura sair de uma faixa de conforto definida.
- Usar os botões A e B para registrar eventos específicos (ex: "Poluição Detectada", "Árvore Plantada") juntamente com as leituras ambientais.
- Adicionar um sensor de vento externo (usando os pinos de I/O) para uma estação meteorológica mais completa.
- Criar um mini-ambiente controlado (caixa de papelão com plástico) e monitorar a temperatura e luz dentro e fora para entender o efeito estufa.
- Para registro de dados a longo prazo, seria necessário adicionar um módulo de cartão SD via pinos de expansão e programar para salvar as leituras.

Projeto 3.4: Registrador de Dados (Extra)

Placa Utilizada: Micro:bit

A extensão **Registrador de Dados** (*Data Logger*) grava, na memória flash da Micro:bit V2, séries temporais de valores que você escolher (ex.: nível de luz e temperatura). Depois, basta conectar a placa ao computador, no disco "MICROBIT", abrir o arquivo MY_DATA.HTM, de onde será possível visualizar e copiar os dados para uma planilha, ou baixar em CSV e construir gráficos no Excel.

Objetivo de Aprendizagem:

- **Matemático:** Construir tabela tempo \times luminosidade \times temperatura e escolher o tipo de gráfico adequado (linha/dispersão). Interpretar gráficos: tendência, máximos/mínimos, intervalos de estabilidade. Calcular média, mediana, amplitude e desvio padrão das séries. Estimar taxa de variação e relacioná-la à inclinação do gráfico. Ajustar um modelo simples (linear ou exponencial) e avaliar qualidade do ajuste.
- **Computacional:** Usar variáveis e entradas de sensores (nível de luz, temperatura) com nomes e unidades claros. Decompor o problema "coletar \rightarrow armazenar \rightarrow transferir \rightarrow analisar" e explicar o fluxo de dados. Exportar CSV e documentar o experimento para reprodutibilidade (código, parâmetros e contexto).

Conceitos Matemáticos Envolvidos:

- **Aplicar Conceitos de Estatística:** Praticar a coleta, organização e interpretação de dados reais.
- **Construir Gráficos:** Visualizar as variações ambientais através de gráficos de linha ou de barras.
- **Compreender Grandezas e Medidas:** Entender como a tecnologia pode medir temperatura e luz e representar essas grandezas numericamente.

- **Desenvolver Pensamento Computacional:** Ver na prática como a programação e os sensores são usados para coletar informações do mundo físico.
- **Conscientização Ambiental:** Utilizar os dados para discutir questões como microclimas na escola, eficiência energética e a importância do monitoramento ambiental.

Materiais Necessários:

- 1 BBC Micro:bit
- 2 Pilhas AAA
- Suporte para Baterias AAA

Passo a Passo da Montagem do Circuito:

Assim como em projetos anteriores com uso da Micro:bit, este projeto se beneficia dos sensores integrados da placa. A montagem é mínima, isto é, basta conectar a Micro:bit ao computador via cabo USB.

Como programamos (no MakeCode)

1. Adicionar a extensão "Registador de dados".
2. No bloco no iniciar, combine os blocos:
 - set timestamp → minutes (ou seconds)
 - set columns → "luminosidade", "temperatura"
3. No laço every 60 000 ms (1 min), combine os blocos:
 - log data → coluna "luminosidade" = nível de luz
 - log data → coluna "temperatura" = temperatura (°C)
4. Carregar o código para a Micro:bit e deixá-la no local seguro e pré-definido, para que ocorra a captura dos dados, em um intervalo de tempo satisfatório.

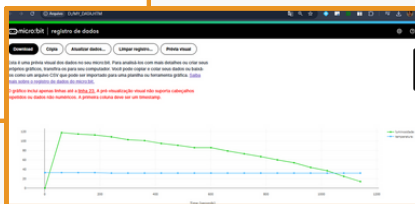
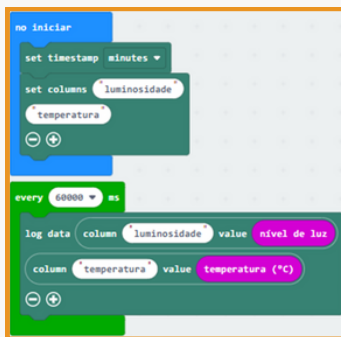
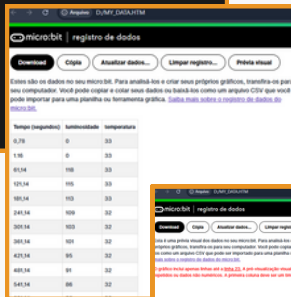
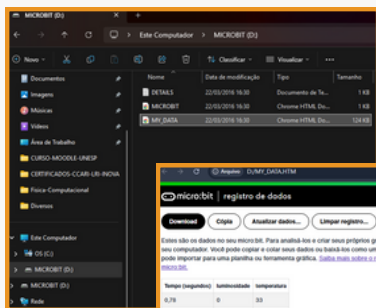
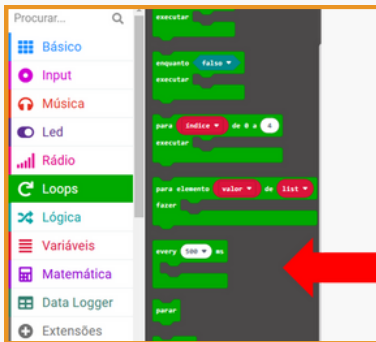
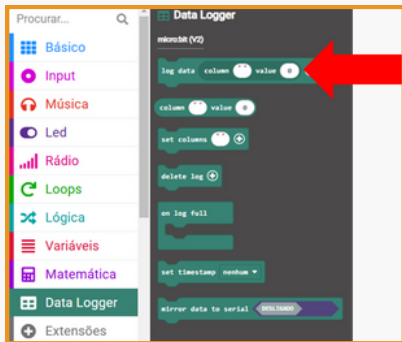
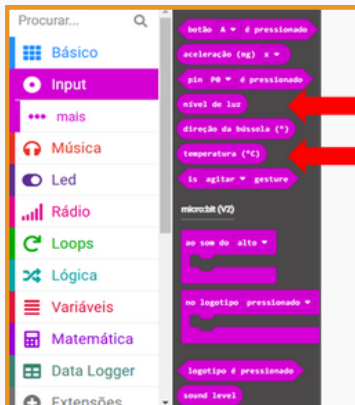
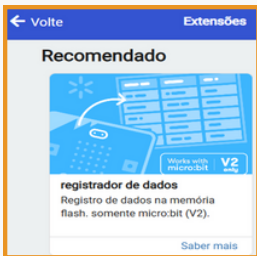
Como retiramos os dados

- Após os dados coletados, conectar a micro:bit ao PC → abrir a unidade MICROBIT → MY_DATA.HTM.
- Clicar em Download (CSV) ou Cópia/Prévia visual.
- Importar o CSV para Excel e construir os gráficos.

No MakeCode:

Materiais:

- 1 BBC Micro:bit
- 2 Pilhas AAA
- Suporte para Baterias AAA



Relevância deste experimento:

- **Dá contexto real:** os alunos capturam luminosidade/temperatura com a Micro:bit e veem dados “vivos”, conectando matemática ao mundo físico e às ciências.
- **Ciclo completo de investigação:** formular pergunta → escolher intervalo (amostragem) → coletar → organizar → analisar → comunicar (tabelas/gráficos/relatório).
- **Pensamento computacional na prática:** programar o registro periódico, usar carimbo de tempo, exportar para planilha e automatizar cálculos.
- **Letramento de dados:** discutir variáveis, outliers, médias/medianas e leituras críticas de gráficos – exatamente o que a BNCC pede para os Ensino Fundamental e Médio.

Competências e Habilidades da BNCC:

Ensino Fundamental

Competências específicas: investigar, organizar, representar e comunicar informações; usar tecnologias digitais para modelar e resolver problemas; articular estatística/probabilidade com outras áreas e com pensamento computacional.

Habilidades por anos (onde o experimento se aplica):

- **4º ano:** registrar temperaturas e representar variações em gráficos de colunas; analisar tabelas/gráficos e relatar conclusões. (EF04MA24, EF04MA27, EF04MA28).
- **7º ano:** planejar e realizar experimentos aleatórios; calcular e interpretar média (relacionando à amplitude); planejar pesquisa e comunicar por relatório com tabelas/gráficos; interpretar gráficos de setores. (EF07MA34, EF07MA35, EF07MA36, EF07MA37).

- **8º ano:** calcular probabilidades via espaço amostral; escolher o tipo de gráfico adequado; agrupar dados em classes; calcular média, moda, mediana e discutir dispersão; justificar uso de amostras e executar pesquisa amostral com relatório. (EF08MA22–EF08MA27).
- **9º ano:** analisar gráficos da mídia (escalas/legendas inadequadas); escolher e construir o melhor gráfico; planejar e executar pesquisa amostral com medidas de tendência central e amplitude. (EF09MA21–EF09MA23).

Ensino Médio (Matemática e suas Tecnologias)

Competências específicas mobilizadas: interpretar situações com dados; modelar e resolver problemas usando diferentes registros (estatístico, computacional); investigar com tecnologias e argumentar criticamente (inclusive sobre amostragem e representações na mídia).

Habilidades diretamente trabalhadas:

- Leitura crítica de dados: analisar tabelas/gráficos e identificar inadequações (EM13MAT102).
- Pesquisa amostral com tecnologia: planejar e executar pesquisa; relatar com médias e dispersão (amplitude, desvio-padrão) – com ou sem recursos digitais (EM13MAT202).
- Medidas de tendência central e dispersão em diferentes contextos (EM13MAT316).
- Contagem e probabilidade (princípios aditivo/multiplicativo; espaço amostral; probabilidades em experimentos sucessivos) (EM13MAT310–EM13MAT312, EM13MAT311).
- Planilhas e automação para organizar/analisar dados (EM13MAT203).

Projeto 3.5: Jogo da Reciclagem

Placa Utilizada: Makey Makey

Objetivo de Aprendizagem:

- **Matemático:** Desenvolver o raciocínio lógico, a classificação de objetos (com base em categorias de reciclagem), a contagem (de acertos/erros) e a compreensão de sequências de eventos e tomada de decisões.
- **Computacional:** Programar interações com o Makey Makey para criar um jogo interativo no Pictoblox, utilizando eventos de teclado, controle de variáveis (pontuação), sons e lógica condicional.

Conceitos Matemáticos Envolvidos:

- **Classificação e Agrupamento:** Categorizar resíduos em seus respectivos tipos de reciclagem (papel, plástico, metal, vidro, orgânico).
- **Contagem e Pontuação:** Atribuir pontos por acertos e subtrair por erros, calcular o total de pontos.
- **Lógica Condicional:** "Se o objeto X for colocado na lixeira Y, então [ação]".
- **Sequência de Eventos:** A ordem das ações no jogo (aparecer objeto, coletar, pontuar, próximo objeto).
- **Probabilidade:** Se os objetos aparecerem aleatoriamente, discutir a chance de aparecer um determinado tipo.

Competências e Habilidades da BNCC:

Este projeto, utilizando o Makey Makey e focado em um jogo, é excelente para o Ensino Fundamental e pode ser estendido para o Ensino Médio com maior complexidade na lógica e análise de dados do jogo.

Para o Ensino Fundamental:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Criar um jogo que simula um processo real (reciclagem), formulando regras e testando a lógica do jogo.
- **4. Comunicação:** Expressar ideias sobre as regras do jogo e a importância da reciclagem através da interação do jogo.
- **5. Cultura Digital:** Utilizar o Makey Makey para transformar objetos em controles de um jogo digital, compreendendo a interação entre o mundo físico e o virtual.
- **9. Empatia e Cooperação:** Trabalhar em grupo para projetar e testar o jogo, além de jogar de forma colaborativa.
- **10. Responsabilidade e Cidadania:** Conscientizar sobre a importância da reciclagem e a correta separação dos resíduos para um ambiente sustentável.

- **Competências Específicas de Matemática:**

- **Números:** Contagem de itens (acertos, erros), pontuação do jogo.
- **Álgebra:** Introdução à lógica de "se... então..." para classificar objetos e atribuir pontos (ex: se for papel, então vá para a lixeira azul e ganhe pontos).
- **Probabilidade e Estatística:** Discutir a ideia de "chance" ao aparecerem diferentes tipos de resíduos no jogo. Coletar dados sobre os tipos de resíduos que mais aparecem.
- **Geometria:** Se o jogo envolver lixeiras visuais, relacionar as formas e posições.

Para o Ensino Médio:

- **Competências Gerais:**

- **2. Pensamento Científico, Crítico e Criativo:** Desenvolver algoritmos de classificação mais sofisticados para os resíduos. Analisar a jogabilidade e propor melhorias.
 - **5. Cultura Digital:** Programar um sistema interativo complexo que responde a múltiplas entradas do Makey Makey, explorando a criação de interfaces lúdicas e educacionais.
 - **6. Trabalho e Projeto de Vida:** Planejar, desenvolver e refinar um projeto de jogo, gerenciando o tempo e os desafios técnicos.
 - **7. Argumentação:** Justificar as escolhas de design do jogo, a lógica de pontuação e as estratégias de engajamento dos jogadores.
 - **10. Responsabilidade e Cidadania:** Aprofundar a discussão sobre gestão de resíduos, economia circular e o papel da gamificação na educação ambiental.
- **Competências Específicas de Matemática e suas Tecnologias:**
 - **Habilidades Relacionadas a Álgebra:**
 - Modelar sistemas de regras e pontuação complexas, utilizando variáveis, expressões e condicionais aninhadas.
 - Explorar a lógica booleana avançada para classificação e validação de entradas.
 - **Habilidades Relacionadas à Probabilidade e Estatística:**
 - Analisar a distribuição de tipos de resíduos que aparecem no jogo.
 - Coletar e analisar dados de desempenho dos jogadores (taxa de acertos, tempo por jogada) para otimizar a dificuldade do jogo.
 - Calcular probabilidades de eventos específicos no jogo.
 - **Habilidades Relacionadas a Funções:**
 - Compreender a entrada do Makey Makey como uma função que mapeia um toque a uma ação ou valor no jogo.

Materiais Necessários:

- 1 Placa Makey Makey
- 1 Cabo USB para Makey Makey
- 5 Cabos jacaré (preto, verde, branco, vermelho e amarelo)
- 1 Caixa de papelão pequena
- Massinha de modelar nas cores verde, branco (ou azul), vermelho e amarelo
- 4 percevejos latonados (para melhor fixação dos cabos jacarés nas massinhas de modelar)
- 1 Computador com Pictoblox instalado e com internet

Este jogo será construído totalmente virtual, isto é, no Pictoblox. Mas, também poderá ser vinculado à lixeirinhas físicas, feitas em papelão (rolo de papel higiênico), que podem ser adaptadas do jogo desenvolvido pelo Blog da Robótica, chamado Reciclagem com Arduino e Sucata, disponível em <https://www.blogdarobotica.com/2023/05/12/semana-do-scratch-reciclagem-com-arduino-e-sucata/>.

Passo a Passo da Montagem do Circuito:

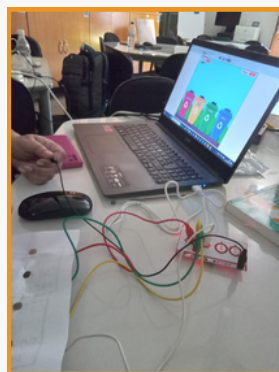
1. Conecte a Makey Makey: Conecte a Makey Makey ao seu computador via cabo USB. O computador o reconhecerá como um teclado/mouse padrão.

2. Conecte o "terra": Use o cabo jacaré preto para conectar o pino terra da Makey Makey a algo que você ou o usuário irá segurar constantemente. Isso pode ser uma pulseira ajustada ao corpo, um anel, um pedaço de massinha que a pessoa segure, ou a própria mão (segurando a ponta do cabo). Este é o ponto de referência do circuito.

3. Conecte os Sensores:

- Enrole cada percevejo latonado em um pouco de massa de modelar, sendo cada uma de cor diferente. Deixe a ponta do percevejo à mostra, para fixação dela na caixa de papelão e, posteriormente, para fixação do cabo jacaré.
- Monte a caixa de papelão conforme indicando na imagem, observando os pontos da distribuição e conexão da massinha de modelar.

- Conecte cada uma das pontas dos cabos jacarés a somente uma das pontas de cada percebejo, observando a respectiva cor da massinha, a cor do cabo e, a lixeira que ele será referenciado.
- Conecte a a outra ponta dos cabos jacarés, em uma das entradas de "tecla" no Makey Makey: seta para direita (jacaré vermelho), seta para esquerda (jacaré branco ou azul), seta para cima (jacaré amarelo), seta para baixo (jacaré verde).



Passo a Passo da Programação no Pictoblox:

- 1. Abra o Pictoblox:** Inicie o software Pictoblox em seu computador.
- 2. Selecione o Dispositivo Makey Makey:** No menu "Dispositivos", selecione "Makey Makey".
- 3. Adicionar Extensões:** Certifique-se de ter a extensão "Makey Makey" ativada.
- 4. Crie as variáveis "Pontos" e "Tempo":** A primeira vai regular a pontuação de cada jogador e, a segunda, vai limitar o tempo de jogo para cada participante.

5. Crie o Sprite da Lixeira: Crie sprite para a "lixeyras", programando-o para aparição no jogo em dois momentos. Uma no início do jogo e outra no final (Game Over).

6. Crie os Sprites dos resíduos: Crie sprites para cada um dos "resíduos" (plástico, metal, vidro e papel), que aparecerão no jogo de forma aleatória em que, cada um, terá sua própria programação.

7. Lógica do Jogo:

- Inicialização: quando [bandeira verde] clicada:
 - seta pontuação para 0.
 - seta tempo para 30 segundos
 - mostra as lixeiras no primeiro cenário.
 - esconde todos os resíduos.
 - transmite [jogo iniciado].
 - sorteia um resíduo aleatório e faz ele se deslocar para baixo.
- Lógica de Classificação e Pontuação (para cada resíduo):
 - quando a tecla [seta para esquerda] for pressionada (ex: lixeira de plástico):
 - se (Objeto Atual) = [ID do plástico] (ex: 1):
 - o resíduo vai para a lixeira correspondente.
 - muda pontuação por 1 unidade.
 - toca o som [acerto].
 - chama o bloco "sorteia um resíduo aleatório", fazendo ele se deslocar para baixo.
 - senão:
 - muda pontuação por -1 unidade.
 - toca o som [erro].
 - chama o bloco "sorteia um resíduo aleatório", fazendo ele se deslocar para baixo.
 - Repita a lógica para cada resíduo e tecla da Makey Makey correspondente (seta para direita, seta para cima e seta para baixo).

8. Exibir Tempo e Pontuação: Ao final do tempo de 30 segundos, no cenário 2 (Game Over) das lixeiras, será exibido a "Pontuação" do jogador e o "Tempo" igual a zero, indicando o fim do jogo.

Programação no Pictoblox dos Resíduos:

```
quando for clicado
  mude Pontos para 0
  mude Tempo para 30
  transmite jogo iniciado
  define o tamanho como 25 %
  mostre
  mude para a fantasia número aleatório entre 1 e 4
  vá para posição aleatória
  mude y para 180
  sempre
    adicione -6 a y
    espere 0.125 seg
```



```
quando eu receber jogo iniciado
  espere 5 seg
  sempre
    se Pontos = 0 então
      esconda
      pare todos
```

```
quando eu receber jogo iniciado
  espere 1 seg
  sempre
    repita até que Tempo = 0
      espere 1 seg
      adicione -1 a Tempo
  esconda
  pare todos
```

OBS: as teclas da Makey Makey devem ser programadas na área dos resíduos, haja visto que estas responderão à ações para este ator.

```
quando a tecla seta para esquerda for pressionada
  deslize por 1 segs. até x: -177 y: 1

quando a tecla seta para direita for pressionada
  deslize por 1 segs. até x: 177 y: 1
```

```
quando a tecla seta para cima for pressionada
  deslize por 1 segs. até x: -55 y: 1

quando a tecla seta para baixo for pressionada
  deslize por 1 segs. até x: 55 y: 1
```


Programação no Pictoblox das Lixeiras:



Testando o Projeto:

Com a Makey Makey conectada e o código carregado no Pictoblox:

- Um "resíduo" aparecerá na tela.
- O jogador (segurando o "Terra") deverá tocar a "tecla" (na massinha da respectiva lixeira) correspondente à categoria correta do resíduo.
- Se acertar, ganha pontos e um novo resíduo aparecerá.
- Ou, se errar, perde pontos e um novo resíduo aparece.
- O jogo continua, desafiando a velocidade e o conhecimento do jogador sobre reciclagem e classificação.
- Ao final de 30 segundos, o jogo finalizará automaticamente e, se tiver participando mais de um jogador, ganhará o que tiver o maior número de acertos. Isto é, ganhará o que reciclar/descartar o lixo corretamente.

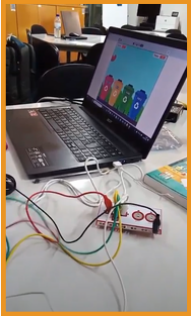
Sugestões de Aplicação em Sala de Aula de Matemática:

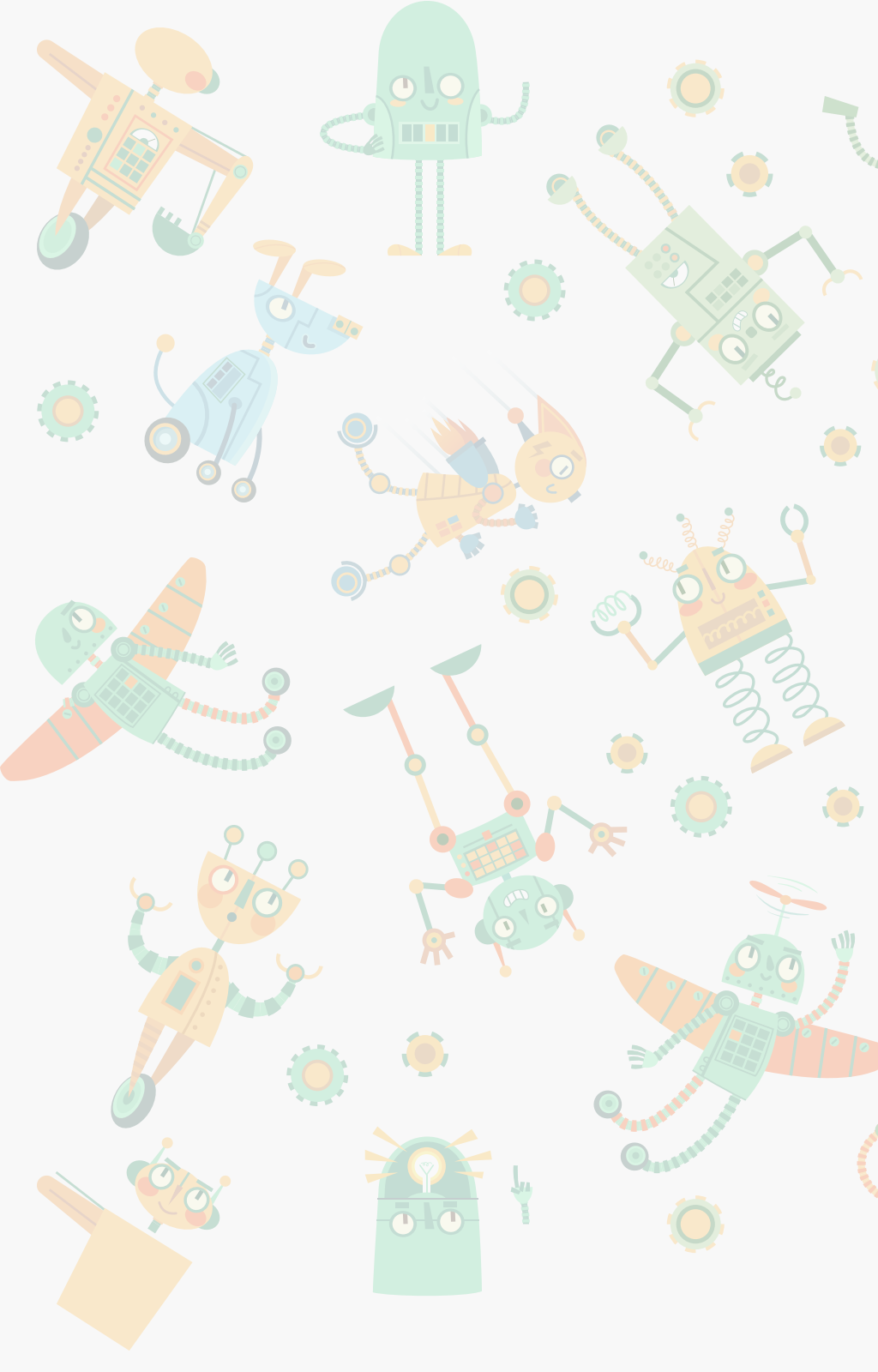
- **Contagem e Operações Fundamentais:** Praticar contagem de acertos e erros. Cálculos de pontuação, incluindo números negativos.
- **Classificação e Conjuntos:** Categorizar objetos com base em suas propriedades de reciclagem, o que se relaciona com conceitos de conjuntos e subconjuntos.
- **Probabilidade:** Analisar a frequência com que cada tipo de resíduo aparece e discutir as probabilidades. Calcular a probabilidade de acertar ou errar um item.
- **Porcentagens e Taxas:** Calcular a taxa de acerto do jogador ao final da partida.
- **Gráficos de Desempenho:** Coletar os dados de pontuação dos alunos e criar gráficos de barras para comparar o desempenho da turma.
- **Lógica Proposicional:** Para alunos mais avançados, discutir a lógica por trás da classificação: "Se (material = plástico) E (cor = verde), então (lixeira = plástico verde)".

Desafios/Variações:

- **Níveis de Dificuldade:** Aumentar a velocidade de aparecimento dos objetos ou introduzir mais categorias de resíduos.
- **Feedback Visual:** Além do som, fazer com que as lixeiras mudem de cor ou exibam uma animação de "correto/incorreto".
- **Ranking de Pontuação:** Criar um sistema de ranking para os melhores jogadores da turma.
- **Modo Cooperativo/Competitivo:** Adaptar o jogo para ser jogado em duplas ou grupos, fomentando a competição saudável ou a colaboração.
- **Simulação de Impacto:** Adicionar um contador de "lixo não reciclado" que cresce, mostrando visualmente o impacto do descarte incorreto.

Alguns registros do 3º. Encontro:





CAPÍTULO 4

A VOZ DA EXPERIÊNCIA - ANÁLISE DO CURSO DE FORMAÇÃO



Análise do Curso de Formação

Este capítulo se dedica à análise dos dados coletados por meio dos formulários de Avaliação Diagnóstica e Avaliação de Satisfação aplicados aos participantes do curso de formação em "Robótica Educacional com Metodologias Ativas" . O objetivo é compreender o perfil dos professores, suas expectativas iniciais, o nível de conhecimento prévio, e a percepção de satisfação e aprendizado ao final do curso, validando a eficácia da proposta pedagógica e das metodologias empregadas.

4.1. Metodologia de Análise de Dados: Uma Abordagem Qualitativa

A análise dos dados provenientes dos formulários de Inscrição, de Avaliação Diagnóstica e de Satisfação do curso, foi conduzida sob uma perspectiva de pesquisa qualitativa. Este tipo de abordagem é fundamental quando se busca compreender fenômenos complexos em seu contexto natural, explorando as percepções, experiências e significados atribuídos pelos participantes, em vez de focar apenas em números e estatísticas.

Na obra "Pesquisa qualitativa do início ao fim", Yin (2016) destaca que a pesquisa qualitativa é uma jornada que envolve a coleta de dados de múltiplas fontes, a análise aprofundada desses dados para identificar padrões e temas emergentes, e a interpretação rica para construir uma compreensão holística do fenômeno estudado.

No contexto deste trabalho, a opção pela metodologia qualitativa permitiu ir além da mera quantificação de respostas "sim" ou "não", ou de escalas de satisfação. Ela possibilitou a exploração das nuances presentes nas respostas abertas dos professores, bem como a interpretação das escolhas em questões fechadas à luz do contexto geral do curso e do perfil dos participantes.

O processo de análise seguiu os seguintes passos, alinhados aos princípios da pesquisa qualitativa:

- **Coleta de Dados:** Utilização de formulários de questionário com perguntas fechadas (escalas de satisfação e níveis de conhecimento) e abertas (expectativas e sugestões), aplicados em momentos distintos do curso. Os dados foram considerados como a "voz" direta dos professores.
- **Organização e Preparação:** As respostas foram compiladas e organizadas. Para as questões abertas, foi realizada uma leitura inicial para familiarização com o conteúdo e identificação de temas recorrentes e pontos de destaque.
- **Análise Temática/Conteúdo:** A análise das respostas abertas, em particular, baseou-se na identificação de temas, categorias e padrões emergentes. Termos, frases e ideias semelhantes foram agrupados, permitindo sintetizar as percepções e sugestões dos professores. Por exemplo, a partir de diversas frases sobre "dificuldade em montar", "não funcionou de primeira" e "precisa de ajuda com os fios", emergiu o tema da "necessidade de aprofundamento em eletrônica".
- **Interpretação e Contextualização:** Os temas e padrões identificados foram então interpretados à luz do referencial teórico do curso (BNCC, PC, ABP, RE) e das metodologias e ferramentas empregadas. A análise buscou compreender o "porquê" por trás das respostas, correlacionando o conhecimento prévio, as expectativas e os resultados de satisfação.

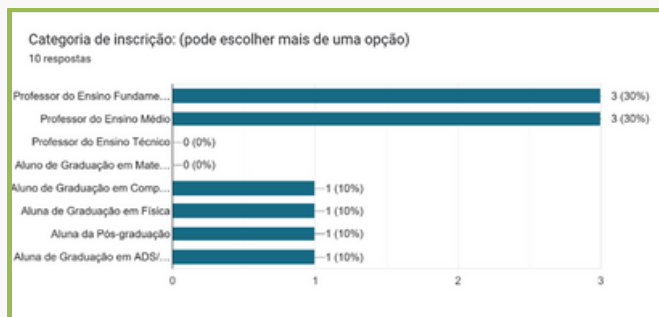
A abordagem qualitativa foi crucial para entender a complexidade da experiência de formação docente, permitindo que as vozes dos professores guiassem a compreensão sobre os impactos reais do curso em sua prática e formação. Com base nas informações coletadas no formulário de inscrição, foi possível traçar um perfil profissional geral dos professores participantes, preservando o sigilo dos dados pessoais.

4.2. Perfil Profissional e Experiência Prévia dos Participantes

O curso foi direcionado a professores de matemática do Ensino Fundamental (EF) e do Ensino Médio (EM). A análise do **Formulário de Inscrição** revelou um grupo heterogêneo em termos de experiência docente geral, mas com um ponto comum: a busca por inovação e aprimoramento profissional na área de tecnologia e metodologias ativas.

Muitos expressaram o desejo de "adquirir conhecimento prático" e "aplicar o aprendido com futuros alunos", demonstrando uma forte motivação para integrar a robótica e a programação em suas práticas pedagógicas.

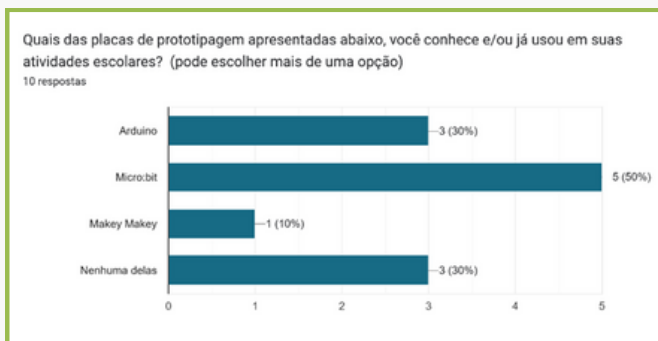
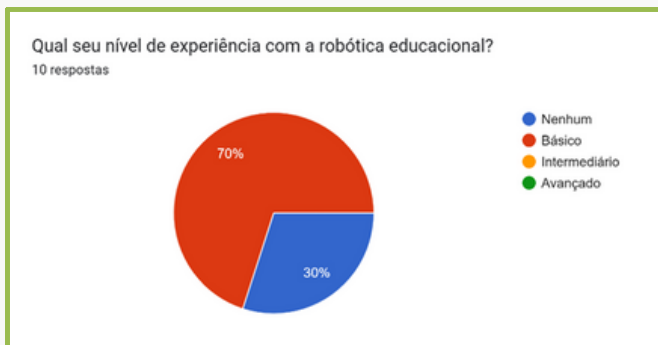
Mesmo com a maioria dos professores atuando no Ensino Fundamental e/ou Médio, sendo a Matemática a disciplina principal de sua carga horária, alguns docentes, pela necessidade da escola, também atuavam em outras disciplinas afins. Essa diversidade contribuiu para um ambiente rico em diferentes realidades e desafios educacionais.

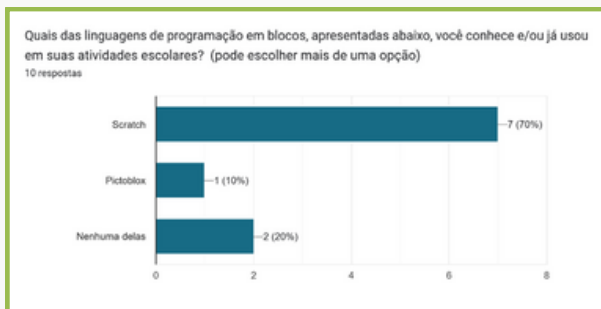


Com relação à experiência prévia com tecnologia e programação, o cenário era heterogêneo, porém com uma predominância de iniciantes ou com pouco contato formal:

- **Nenhum ou Pouco Contato com Robótica Educacional:** Muitos indicaram que o curso seria sua primeira imersão prática em robótica educacional com placas como Arduino Uno, Micro:bit e Makey Makey.

- **Familiaridade com Ferramentas Digitais Gerais:** Uma parte dos professores já utilizava ferramentas digitais em suas aulas (softwares de apresentação, planilhas, plataformas de ensino online), mas sem aprofundamento em programação ou eletrônica.
- **Alguns com Experiência com Scratch ou Linguagens Similares:** Uma minoria já havia tido algum contato com plataformas de programação em blocos como o Scratch, o que lhes dava uma pequena vantagem na compreensão inicial do Pictoblox.
- **Expectativa em relação à eletrônica:** Como visto nas expectativas, a eletrônica era uma área de grande interesse, mas também de pouca experiência prévia para a maioria.





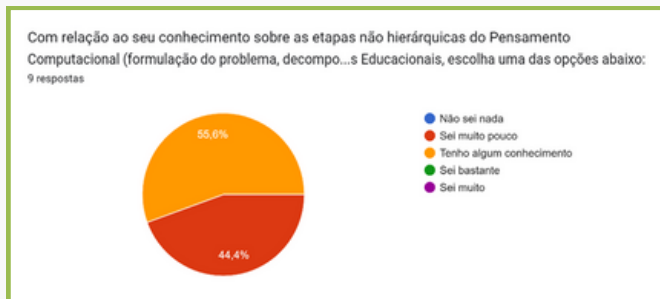
Este perfil inicial reforça a necessidade de o curso ser inclusivo, partindo do básico e construindo o conhecimento de forma gradual, garantindo que todos os professores, independentemente de sua experiência anterior, pudessem se engajar e se sentir capazes de utilizar as tecnologias propostas.

4.3. Avaliação Diagnóstica: Um Panorama Inicial

A **Avaliação Diagnóstica** foi aplicada no início do curso para mapear o conhecimento prévio dos professores em relação aos temas centrais que seriam abordados. A análise das respostas permite contextualizar o ponto de partida dos participantes e entender suas principais expectativas.

4.3.1. Conhecimento sobre as Etapas do Pensamento Computacional

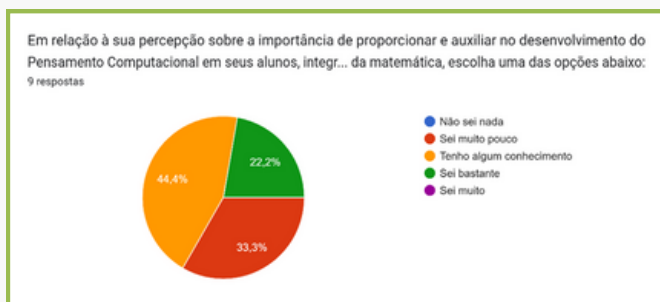
A primeira questão buscou avaliar o conhecimento dos participantes sobre as etapas não hierárquicas do Pensamento Computacional (formulação do problema, decomposição, reconhecimento de padrões, abstração, produção de algoritmos e depuração) e sua aplicação em atividades educacionais.



Estes resultados revelam um cenário em que a maioria dos professores inicia o curso com uma base limitada sobre o Pensamento Computacional. A concentração de respostas nas categorias de "não sei nada" ou "sei muito pouco" sublinha a urgência e a pertinência de um curso que ofereça uma fundamentação sólida nesse campo. Fica evidente que a proposta de desmistificar o Pensamento Computacional, apresentando-o de forma aplicada e não restrita à ciência da computação (como preconizado por Wing, 2006, 2011), é crucial para este público, que busca integrar essas habilidades em sua prática pedagógica.

4.3.2. Percepção sobre a Importância do Pensamento Computacional na Matemática

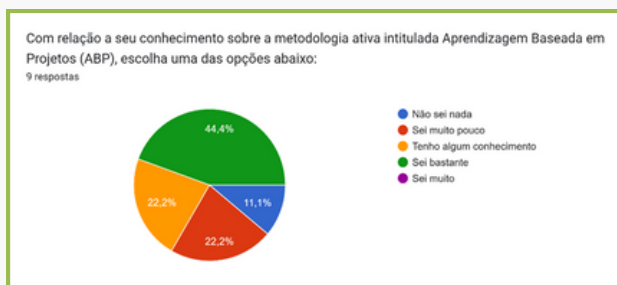
A segunda questão explorou a percepção dos professores sobre a importância de proporcionar e auxiliar no desenvolvimento do Pensamento Computacional em seus alunos, integrando-o ao processo de aprendizagem da matemática.



É interessante notar que, embora o conhecimento técnico sobre as etapas do Pensamento Computacional fosse limitado, a percepção de sua importância para a matemática também era incipiente para uma parcela significativa dos participantes. Isso indica que a demanda por clareza sobre o porquê de se integrar o pensamento computacional à matemática é tão relevante quanto o como. O curso precisou, portanto, não apenas ensinar os fundamentos práticos, mas também construir a argumentação pedagógica que conecta profundamente essas duas áreas do conhecimento, mostrando que o Pensamento Computacional é uma competência transversal que enriquece a resolução de problemas matemáticos.

4.3.3. Conhecimento sobre Aprendizagem Baseada em Projetos (ABP)

Esta questão avaliou a familiaridade dos participantes com a metodologia ativa de Aprendizagem Baseada em Projetos (ABP).

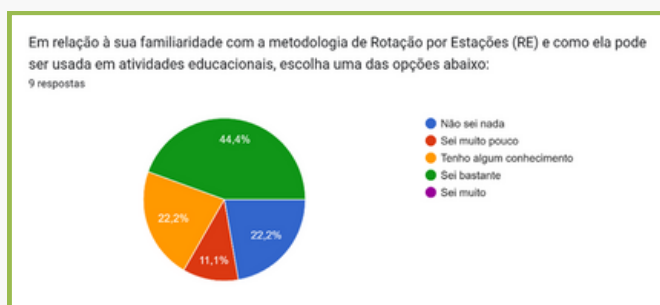


Os dados demonstram que a Aprendizagem Baseada em Projetos (ABP) era uma metodologia majoritariamente desconhecida pelos professores que iniciaram o curso, com 66,6% (somando "não sei nada" e "sei muito pouco") reportando pouca ou nenhuma familiaridade. Este cenário validou a decisão de dedicar uma parte substancial da formação à explicação conceitual e, principalmente, à vivência prática da ABP.

A necessidade de contextualizar a aprendizagem da matemática e de engajar os alunos em problemas reais torna a ABP uma ferramenta pedagógica de grande valor, e os resultados apontam para uma lacuna na formação contínua dos professores nessa área.

4.3.4. Familiaridade com Rotação por Estações (RE)

A questão seguinte focou na familiaridade com a metodologia de Rotação por Estações (RE) e sua aplicabilidade em atividades educacionais.



A Rotação por Estações (RE) apresentou um perfil de conhecimento inicial muito similar ao da ABP, com 77,7% dos professores expressando pouca ou nenhuma familiaridade. Isso é compreensível, dado que a RE é frequentemente utilizada como um modelo de implementação para a ABP, especialmente em ambientes que utilizam múltiplas ferramentas e exigem flexibilidade didática. A vivência prática proporcionada pelo curso, estruturada exatamente com base nas estações, foi, portanto, um componente didático essencial. Ela permitiu que os professores não apenas entendessem o conceito da RE, mas também experimentassem na prática como organizar e gerenciar diferentes atividades e tecnologias em um mesmo espaço-tempo de aprendizagem.

4.3.5. Expectativas para o Curso

O formulário diagnóstico também incluiu uma questão aberta sobre as expectativas dos participantes para o curso. As respostas foram variadas, mas convergiram para alguns temas principais:

- **Conhecimento prático e aplicação:** Muitos expressaram o desejo de "Obter conhecimento prático dos conteúdos e conseguir aplicá-los em projetos futuros" e "Compreender as principais funções das placas embarcadas, a conexão entre elas e sua consequente aplicação em projetos educacionais".
- **Utilização em sala de aula:** Houve forte expectativa em "Aprender essa forma simples de trabalhar com robótica, para utilizá-la futuramente com alunos" e "Aprimorar na robótica, principalmente na parte eletrônica".
- **Associação com a Matemática:** Alguns professores, especificamente, esperavam "desenvolver meus conhecimentos e habilidades da robótica educacional, a fim de facilitar a associação do tema com o ensino de matemática".
- **Ampliação de repertório e troca de experiências:** A busca por "ampliar meu repertório para contribuir com a minha prática" e "trocar experiências com base nas vivências obtidas ao longo dos anos como professora" também foi recorrente.

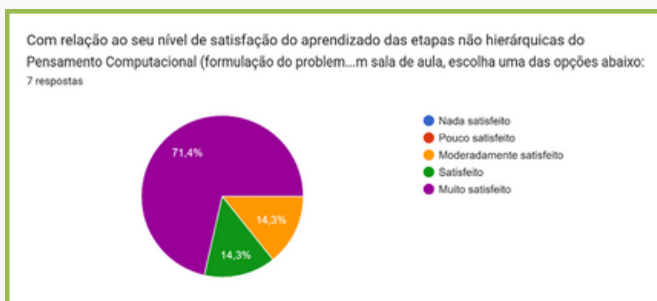
As expectativas iniciais dos professores estavam fortemente alinhadas com a proposta central do curso. Eles buscavam uma formação que fosse além da teoria, focada na aquisição de conhecimentos práticos em robótica educacional, com uma clara intenção de aplicar o aprendizado em suas salas de aula. A ênfase na "associação com a matemática" e na "ampliação de repertório" demonstra uma percepção aguçada sobre a necessidade de inovar no ensino da disciplina e de se manterem atualizados. A valorização da "troca de experiências" também ressalta a importância do ambiente colaborativo e presencial do curso, onde o aprendizado entre pares é tão significativo quanto o conteúdo transmitido.

4.4. Avaliação de Satisfação: O Impacto do Curso

A Avaliação de Satisfação foi aplicada ao final do curso, permitindo coletar a percepção dos participantes sobre o aprendizado, a organização do material, a relevância dos projetos e a experiência geral. As respostas foram cruciais para medir o impacto da formação e identificar pontos de sucesso e de aprimoramento.

4.4.1. Satisfação com o Aprendizado do Pensamento Computacional

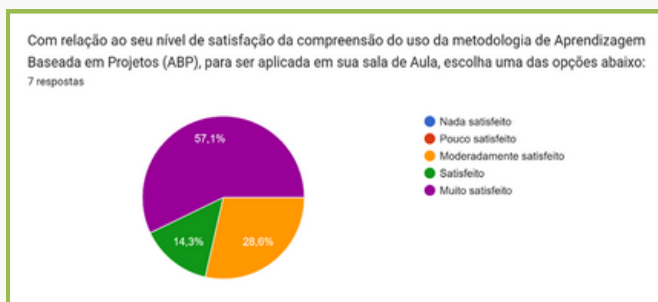
Esta questão avaliou o nível de satisfação do aprendizado das etapas não hierárquicas do Pensamento Computacional e sua aplicação em sala de aula.



A melhora significativa na satisfação em relação ao Pensamento Computacional, especialmente quando comparada ao baixo conhecimento inicial dos participantes, é um dos maiores indicadores de sucesso do curso. Os 71,4% de professores que se declararam "Satisfeitos" ou "Muito Satisfeitos" demonstram que a abordagem prática e contextualizada foi eficaz em desmistificar o Pensamento Computacional e em mostrar sua aplicabilidade direta na educação, especialmente na matemática. Isso sugere que a metodologia do curso conseguiu não apenas apresentar os conceitos, mas também gerar confiança e entusiasmo nos professores para integrar essa competência em suas práticas pedagógicas.

4.4.2. Satisfação com a Compreensão da Aprendizagem Baseada em Projetos (ABP)

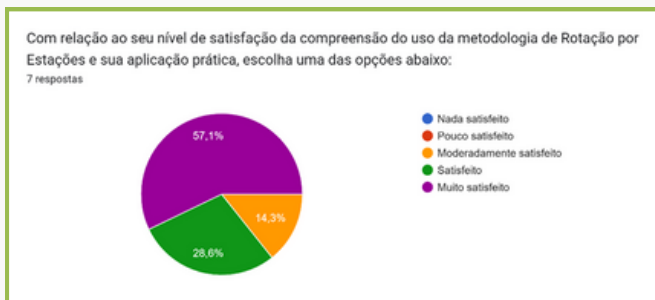
A questão sobre a compreensão da metodologia de Aprendizagem Baseada em Projetos (ABP) para ser aplicada em sala de aula, também, trouxe uma resposta significativa.



Apesar de ser uma metodologia pouco conhecida no início do curso, mais da metade dos participantes expressou satisfação com a compreensão da ABP. Isso valida a abordagem do curso em utilizar a Aprendizagem Baseada em Projetos como a espinha dorsal de suas atividades. A vivência de projetos desde a problematização até a prototipagem, permitindo que os professores experimentassem o ciclo completo da ABP, foi fundamental para que eles compreendessem seu valor intrínseco e seu potencial para engajar os alunos e contextualizar o aprendizado. A percepção de que a metodologia é aplicável em sala de aula é um resultado crucial para a formação docente.

4.4.3. Satisfação com a Compreensão da Rotação por Estações (RE)

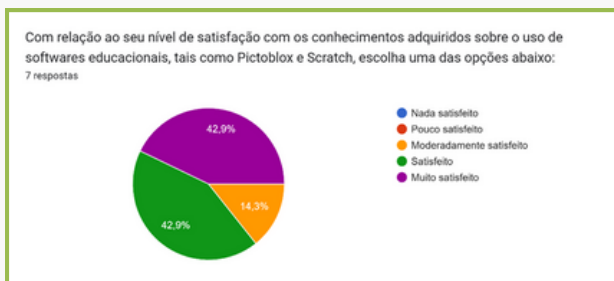
Esta questão avaliou a satisfação com a compreensão do uso da metodologia de Rotação por Estações (RE) e sua aplicação prática.



A satisfação com a metodologia de aprendizado por meio da Rotação por Estações, que foi o formato de organização dos encontros presenciais, também obteve um alto índice de aprovação. O fato de os professores terem vivenciado a metodologia em primeira mão, experimentando a dinâmica de transitar entre diferentes focos de aprendizado (as estações de cada uma das placas), provavelmente contribuiu para essa alta satisfação. Isso demonstra que a RE não é apenas uma estratégia eficaz para o ensino com tecnologias, mas também uma metodologia que os educadores podem compreender e replicar com confiança em suas próprias salas de aula.

4.4.4. Satisfação com os Conhecimentos Adquiridos sobre Softwares Educacionais (Pictoblox e Scratch)

A questão sobre a satisfação com os conhecimentos adquiridos sobre softwares educacionais, tais como Pictoblox e Scratch, nos mostram a evolução da aprendizagem por parte dos docentes, que a priori, não tinham contato com a linguagem de programação em blocos.



A satisfação com o aprendizado dos softwares de programação em blocos (Pictoblox e Scratch) apresentou uma distribuição mais heterogênea em comparação com as metodologias, com uma parcela notável de "Nada satisfeito". Isso pode sugerir que, apesar da interface amigável do Pictoblox, a profundidade ou a agilidade no uso dessas plataformas ainda representa um desafio para uma parte dos professores. É possível que o tempo dedicado à exploração dos softwares não tenha sido suficiente para todos, ou que a necessidade de praticar a programação de forma autônoma seja maior para consolidar o aprendizado. Esta área aponta para futuras melhorias, como a oferta de mais tempo de prática guiada ou materiais de apoio mais detalhados e específicos para as funcionalidades dos softwares.

4.4.5. Satisfação com o Uso e Aprendizado das Ferramentas Tecnológicas

A satisfação com os conhecimentos adquiridos sobre as ferramentas tecnológicas apresentou resultados variados, refletindo a complexidade e a curva de aprendizado de cada plataforma, especialmente para professores com pouca experiência prévia em eletrônica.

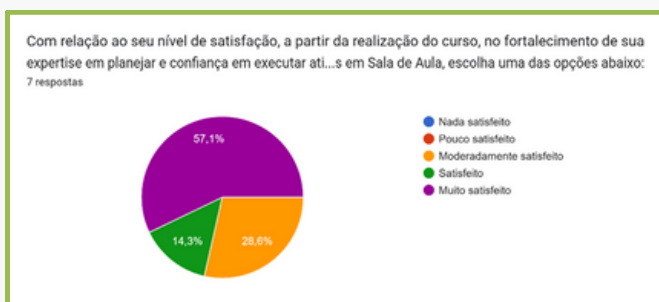


Apesar da menor familiaridade inicial com as placas (conforme avaliação diagnóstica), a satisfação pós-curso foi notável. A soma dos percentuais de satisfação "Satisfeito/Muito Satisfeito", demonstram que uma parte significativa dos professores conseguiu adquirir conhecimentos úteis.

A natureza mais "plug and play" do Makey Makey e a integração de sensores da Micro:bit são pontos fortes que facilitaram o primeiro contato com essas placas mas, que não diminuíram a vontade do aprendiz do da montagem dos circuitos dos projetos com Arduino Uno. Mesmo exigindo um pouco mais sobre eletrônica, para um público iniciante, os projetos com esta placa, foram mais complexos, robustos e com resultados extremamente satisfatórios.

4.4.6. Intenção de Aplicação em Sala de Aula e Percepções Gerais

Um dos resultados mais encorajadores da avaliação de satisfação foi a forte intenção de aplicação dos conhecimentos adquiridos:



Este dado qualitativo demonstra o impacto transformador do curso, validando a abordagem prática e a relevância percebida pelos participantes.

4.4.7. Sugestões de Melhorias para Próximas Edições (Questão Aberta)

O formulário de satisfação também permitiu que os participantes deixassem sugestões e comentários abertos, que são extremamente valiosas para o aprimoramento contínuo do curso. As principais sugestões foram:

- Muitos destacaram a abordagem prática e a oportunidade de "colocar a mão na massa" como o ponto mais positivo. Frases como "a parte prática e de montagem dos projetos foi a melhor" ou "conseguir ver as placas funcionando com meus próprios olhos" foram recorrentes.
- A conexão clara entre a robótica, a programação e os conceitos matemáticos foi outro aspecto muito elogiado, ajudando a "desmistificar a matemática e torná-la mais divertida".
- A metodologia de Rotação por Estações e a possibilidade de experimentar diferentes ferramentas em um ambiente dinâmico também foram apontadas como diferenciais, facilitando a aprendizagem e a troca entre os participantes.
- Professores expressaram que o curso lhes deu "segurança para propor atividades inovadoras" e "ideias para integrar tecnologia nas aulas de forma significativa".

Destacamos também:

- **Aprofundamento em Eletrônica:** A necessidade de "Ter mais subsídios em relação aos conhecimentos mínimos de eletrônica necessários para os projetos, considerando que os três pilares da robótica são: programação, eletrônica e mecânica."
- **Organização e Prazo do Material Online:** A demanda por "Organização do material no google sala de aula; considerando que o curso tem parte online e presencial; as atividades e os pré-requisitos para as aulas precisam ser mais organizados e com prazo maior, fornecidos com a antecedência de um curso a distância."
- **Projetos com Maior Taxa de Sucesso:** A busca por "Projetos que tenham mais chance de dar certo, tenham menos interferências, considerando a heterogeneidade e dimensão de aplicação em sala de aula."
- **Mais Encontros Presenciais e Trocas:** "Acredito que mais encontros presenciais, visto que as trocas são fundamentais para o desenvolvimento das habilidades e construção de conhecimento e rede de parceiros!"

- **Mais Aulas Práticas:** "Adicionar uma ou duas aulas práticas a mais seria interessante para fazer em sala de aula mais experimentos em grupo e participar com os demais na montagem e manuseio dos equipamentos."
- **Exploração de Outras Ferramentas de Programação:** "Sugestão poderia também aplicar em sala de aula, além do Pictoblox, outras ferramentas de programação, mostrando como cada um programa em projetos realizados em sala de aula."

As sugestões sinalizadas pelos participantes do curso, são um rico feedback construtivo. A demanda por maior aprofundamento em eletrônica e projetos mais voltados para a robótica educacional e metodologias ativas, "maker", reflete o desafio da transição entre a teoria e a prática para alguns professores, que podem ter enfrentado dificuldades com a montagem física dos circuitos.

A valorização dos encontros presenciais e aulas práticas reforça o quão crucial é a experiência "mão na massa" e sua interação entre o público docente. A sugestão de explorar outras ferramentas de programação indica um interesse em expandir o repertório, enquanto a questão da organização do material online é um ponto de otimização importante para a experiência híbrida do curso, garantindo que o tempo de estudo a distância seja mais eficiente e produtivo.

4.5. A Trajetória do Aprendizado: Do Diagnóstico à Satisfação Final

A jornada dos professores de matemática neste curso de formação pode ser compreendida como uma progressão significativa, demonstrando uma evolução clara da expectativa inicial e do conhecimento incipiente à satisfação com o aprendizado e à percepção da aplicabilidade.

No início do curso, a Avaliação Diagnóstica, aplicada no Encontro 1, revelou um perfil de participantes com variadas experiências digitais, mas com uma lacuna considerável no uso

prático de tecnologias para a robótica educacional e a programação. A maioria dos professores, atuantes no Ensino Fundamental e/ou Médio, demonstrou pouco ou nenhum conhecimento prévio sobre os pilares do curso: Pensamento Computacional, Aprendizagem Baseada em Projetos (ABP) e Rotação por Estações (RE), bem como sobre as funcionalidades das placas embarcadas (Arduino Uno, Micro:bit, Makey Makey) e softwares como Pictoblox.

Apesar dessa base inicial limitada em conceitos e ferramentas, suas expectativas eram claras e convergentes: buscavam conhecimento prático e relevante, com foco na aplicabilidade imediata em suas salas de aula e, de forma específica, na associação desses novos saberes com o ensino da matemática, além da valorização da troca de experiências com os colegas.

Ao final do curso, a Avaliação de Satisfação (aplicada após o Encontro 3) demonstrou um panorama geral altamente positivo e uma clara transformação no nível de confiança e compreensão dos professores. Houve um aumento notável na satisfação com o aprendizado do Pensamento Computacional (71,4% satisfeitos/muito satisfeitos), da ABP (57,1% satisfeitos/muito satisfeitos) e da RE (57,1% satisfeitos/muito satisfeitos). Isso indica que a metodologia do curso – estruturada com base na Rotação por Estações, aliada à Aprendizagem Baseada em Projetos e focada na experimentação prática com as diferentes placas – foi extremamente eficaz.

A vivência "mão na massa" em cada estação, onde os professores montavam e programavam circuitos para resolver os subproblemas dos temas "Cidades Inteligentes" e "Sensores e Dados para um Ambiente Sustentável", permitiu desmistificar conceitos complexos e torná-los acessíveis, aplicáveis e significativos. Essa capacidade de "aprender fazendo", inerente ao construcionismo, foi essencial para a evolução na percepção e na confiança dos educadores.

Embora a satisfação com o aprendizado dos softwares (Pictoblox e Scratch) tenha sido mais heterogênea (42,9% satisfeitos/muito satisfeitos, com 28,6% nada satisfeitos), as sugestões de melhoria fornecem um roteiro valioso para aprimoramento contínuo. A demanda por maior aprofundamento em eletrônica e a organização do material online apontam para a necessidade de um suporte mais robusto em áreas que ainda geram insegurança ou que precisam de maior autonomia do professor no ambiente híbrido. As sugestões por mais encontros presenciais e aulas práticas reforçam a importância da interação e da experiência coletiva na consolidação do aprendizado.

Em suma, o curso cumpriu seu objetivo de capacitar os docentes, transpondo barreiras de conhecimento prévio e gerando um impacto positivo na percepção e confiança em integrar a robótica educacional e as metodologias ativas ao ensino de matemática. A "voz da experiência" dos professores, coletada por meio desses formulários, não é apenas um atestado de sucesso, mas também um guia essencial para o aprimoramento contínuo, garantindo que futuras formações possam potencializar ainda mais o desenvolvimento de competências e habilidades essenciais para os educadores do século XXI, conforme preconizado pela BNCC.

Finalizamos nossa experiência neste projeto, concluindo que: a Cultura Maker fornece o ambiente de "fazer", o Construcionismo a base teórica para o "aprender fazendo", a Aprendizagem Baseada em Projetos a estrutura para "problemas do mundo real", a Rotação por Estações a "logística" para gerenciar múltiplos projetos e, o Pensamento Computacional o elo de "sinergia" entre todas essas abordagens, que visa a compreensão ativa da Matemática por meio da tecnologia.



REFERÊNCIAS



BACICH, L.; MORAN, J. M. Metodologias ativas para uma educação inovadora. Porto Alegre: Penso, 2018.

BARBOSA, L.; MALTEMPI, M. Matemática, pensamento computacional e BNCC: desafios e potencialidades dos projetos de ensino e das tecnologias na formação inicial de professores. Revista Brasileira de Ensino de Ciências e Matemática, Passo Fundo, v. 3, p. 748-776, ed. espec., 2020.

BENDER, W. N. Aprendizagem baseada em projetos: educação diferenciada para o século XXI. Tradução de De Siqueira Rodrigues. Porto Alegre: Penso, 2014.

BORGHESAN, J. M.; CLEMENT, L. Rotação por estações no ensino de ciências: promovendo suportes à autonomia. Boletim Online de Educação Matemática (BOEM), Florianópolis, v. 11, e0130, 2023.

BRASIL. Base Nacional Comum Curricular. Brasília, DF: MEC, 2017.

CANVA. Plataforma de design gráfico online, 2025. Disponível em: <https://www.canva.com/>.

DO PRADO, S. P.; DANTAS, S. C. Relações entre pensamento computacional e pensamento matemático. Revista Eletrônica de Educação Matemática, Florianópolis, v. 19, p. 1-21, 2024.

HORN, M. B.; STAKER, H.; CHRISTENSEN, C. M. Blended: usando a inovação disruptiva para aprimorar a educação. Porto Alegre: Penso, 2015.

MALTEMPI, M. V. Educação matemática e tecnologias digitais: reflexões sobre prática e formação docente. Acta Scientiae, v. 10, n. 1, p. 59-67, 2008.

PAPERT, S.; HAREL, I. Situating constructionism. In: HAREL, I.; PAPERT, S. (Ed.). Constructionism. Norwood, NJ: Ablex Publishing Corporation, 1991. p. 1-12.

PAPERT, S. The children's machine: rethinking school in the age of the computer. New York: Basic Books, 1994.

PIAGET, J.; GRECO, P. Aprendizagem e conhecimento. Rio de Janeiro: Freitas Bastos, 1974.

PUCCI, M. O.; REICHERT, J. T. Estudo propositivo para aprendizagem significativa das equações algébricas do primeiro grau através do Scratch. *International Journal for Studies in Mathematics Education (JIEM)*, [s. l.], v. 13, n. 3, p. 329-342, 2020.

RIBOLDI, S. M. O.; REICHERT, J. T. Utilização da linguagem de programação Scratch na aprendizagem de funções do 1º grau. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, [s. l.], n. 26, p. 63-71, 2020.

SÁPIRAS, F. S.; VECCHIA, R. D.; MALTEMPI, M. V. Utilização do Scratch em sala de aula. *Educação Matemática Pesquisa*, São Paulo, v. 17, n. 5, p. 973-988, 2015.

VALENTE, J. A. (Org.). *Computadores e conhecimento: repensando a educação*. Campinas: Unicamp/NIEP, 1993.

VALENTE, J. A. *Análise dos diferentes tipos de softwares usados na educação brasileira*. Brasília, DF: Ministério da Educação/Seed, 1999.

VALENTE, J. A. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. *Revista E-curriculum*, v. 14, n. 3, p. 864-897, 2016.

WING, J. M. Computational thinking. *Communications of the ACM*, New York, v. 49, n. 3, p. 33-35, 2006.

WING, J. M. Computational thinking: what and why?. *The Link*, Pittsburgh, v. 6, n. 2, p. 20-21, 2011.

YIN, R. K. *Pesquisa qualitativa do início ao fim*. Tradução de Daniel Bueno. Porto Alegre: Penso, 2016.

Agradecimentos:

- Instituto Federal do Amazonas (IFAM): Edital n°. 05/2023 - DDP/PROGESP/REITORIA/IFAM;
- Fundação de Amparo à Pesquisa do Estado do Amazonas (FAPEAM): Edital n°. 016/2023 - PRODOC/FAPEAM/03.06.2023;
- Instituto de Geociências e Ciências Exatas (IGCE) da Universidade Estadual Paulista (Unesp), campus de Rio Claro.



www.fapeam.am.gov.br

www.ifam.edu.br

www.unesp.br

Agradecimento mais que especial:



www.fapeam.am.gov.br

www.ifam.edu.br

www.unesp.br

