



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

**HENRIQUE DE ALMEIDA MARTINS**

**PLANO DE CURSO PARA PROJETO DE EXTENSÃO: DO BÁSICO AO ROBÔ  
SEGUIDOR DE LINHA**

**RUSSAS**

**2026**

HENRIQUE DE ALMEIDA MARTINS

PLANO DE CURSO PARA PROJETO DE EXTENSÃO: DO BÁSICO AO ROBÔ SEGUIDOR  
DE LINHA

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

Orientador: Prof. Dr. Reuber Regis de Melo.

RUSSAS

2026

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Sistema de Bibliotecas

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

M343p Martins, Henrique de Almeida.  
Plano de Curso para Projeto de Extensão: Do Básico ao Robô Seguidor de Linha / Henrique de Almeida  
Martins. – 2026.  
65 f. : il. color.

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

1. Robótica educacional. 2. Tinkercad. 3. Arduino. I. Título.

CDD 005

---

HENRIQUE DE ALMEIDA MARTINS

PLANO DE CURSO PARA PROJETO DE EXTENSÃO: DO BÁSICO AO ROBÔ SEGUIDOR  
DE LINHA

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

Aprovada em: 23/01/2026

BANCA EXAMINADORA

---

Prof. Dr. Reuber Regis de Melo (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Pablo Luiz Braga Soares  
Universidade Federal do Ceará (UFC)

---

Prof. Ms. Hugo Nathan Barbosa Regis  
Universidade Federal do Ceará (UFC)

Dedico este trabalho à minha mãe, Yanna Marreiro, meu pai, Wellington Martins e minhas avós Cecília Maria e Maria das Graças. Em memória de Helio Rabelo e Raimundo Nonato.

## **AGRADECIMENTOS**

Agradeço primeiramente aos meus pais, Wellington Martins e Yanna Marreiro pelo exemplo e formação como ser humano, além do suporte dado em toda minha vida.

A toda minha família e amigos, pelo apoio e incentivo durante toda minha jornada como graduando.

Ao professor Dr. Reuber Regis de Melo, pela orientação durante todas as etapas deste trabalho de conclusão de curso. Além dos conselhos durante grande parte da minha graduação.

Ao professor Dr. Pablo Luiz Braga Soares pela oportunidade de entrar no projeto EPRA, foram 3 anos de participação e muito aprendizado adquirido.

A todos os professores da universidade que contribuíram na minha formação acadêmica e profissional.

## RESUMO

A Robótica Educacional (RE) vem se consolidando como uma estratégia pedagógica eficaz para o desenvolvimento de competências digitais, raciocínio lógico e criatividade. Para maximizar esses benefícios no ensino superior, este trabalho propõe, aplica e valida um plano de ensino estruturado e progressivo para o ensino de robótica e programação com Arduino, destinado a alunos do ensino superior da Universidade Federal do Ceará (Campus de Russas), através de um curso ofertado pelo projeto de extensão EPRA (Ensino de Programação e Robótica com Arduino) entre os meses novembro e dezembro de 2025. A metodologia desenvolvida inicia com a utilização da plataforma Tinkercad para o aprendizado intuitivo por meio de programação em blocos, realiza uma transição gradual para a sintaxe da linguagem C e exercita os conceitos aprendidos através de desafios práticos, primeiro utilizando robôs ultrassônicos e finalizando com os robôs seguidores de linha. A avaliação da aprendizagem foi contínua, baseada em desafios práticos e questionário que mediram a satisfação e a percepção de dificuldade nos alunos. Os resultados demonstram uma alta taxa de satisfação, redução na percepção de complexidade e efetividade na preparação dos alunos para projetos robóticos com Arduino, validando a abordagem como uma ferramenta educacional acessível e motivadora.

**Palavras-chave:** robótica educacional; tinkercad; arduino.

## ABSTRACT

Educational Robotics (ER) has been consolidating itself as an effective pedagogical strategy for the development of digital competencies, logical reasoning, and creativity. To maximize these benefits in higher education, this work proposes, applies, and validates a structured and progressive teaching plan for robotics and programming with Arduino, aimed at higher education students from the Federal University of Ceará (Russas Campus), through a course offered by the EPRA (Teaching Programming and Robotics with Arduino) extension project between November and December 2025. The developed methodology begins with the use of the Tinkercad platform for intuitive learning through block-based programming, makes a gradual transition to C language syntax, and exercises the learned concepts through practical challenges, first using ultrasonic robots and finishing with line-following robots. Learning assessment was continuous, based on practical challenges and a questionnaire that measured student satisfaction and perception of difficulty. The results demonstrate a high satisfaction rate, a reduction in the perception of complexity, and effectiveness in preparing students for robotic projects with Arduino, validating the approach as an accessible and motivating educational tool.

**Keywords:** educational robotics; tinkercad; arduino.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Placa Arduino Mega e pinagem . . . . .	19
Figura 2 – Modulação de largura de pulso . . . . .	20
Figura 3 – Interface de programação Arduino . . . . .	21
Figura 4 – Página de inscrição do Tinkercad . . . . .	22
Figura 5 – Interface de programação em blocos . . . . .	23
Figura 6 – Interface de programação em texto . . . . .	23
Figura 7 – Sensor ultrassônico . . . . .	24
Figura 8 – Robô ultrassônico . . . . .	25
Figura 9 – Circuito robô seguidor de linha . . . . .	26
Figura 10 – Robô seguidor de linha e indicação de componentes . . . . .	27
Figura 11 – Roteiro de aulas do curso 2023.1 . . . . .	32
Figura 12 – Nível de dificuldade . . . . .	32
Figura 13 – Roteiro de aulas do curso 2023.2 . . . . .	33
Figura 14 – Roteiro de aulas do curso 2024 . . . . .	34
Figura 15 – Roteiro de aulas do curso 2025.2 . . . . .	35
Figura 16 – Instruções para inscrição na classe . . . . .	36
Figura 17 – Circuito e algoritmo do exemplo: Controle de um LED com arduino . . . . .	37
Figura 18 – Algoritmo do exemplo: Sensor de estacionamento com arduino . . . . .	37
Figura 19 – Circuito do exemplo: Sensor de estacionamento com arduino . . . . .	38
Figura 20 – Conversão blocos para código . . . . .	39
Figura 21 – Garagem simulada utilizada no desafio da aula nove . . . . .	40
Figura 22 – Ponte H . . . . .	41
Figura 23 – Robô ultrassônico e indicação de componentes . . . . .	41
Figura 24 – Sensor QTR-8A . . . . .	42
Figura 25 – Robô seguidor de linha e indicação de componentes . . . . .	43
Figura 26 – Distribuição das respostas para as perguntas relacionadas à plataforma Tinkercad. . . . .	46
Figura 27 – Distribuição das respostas para as perguntas relacionadas à lógica de programação em C. . . . .	47
Figura 28 – Distribuição das respostas para as perguntas relacionadas aos componentes e lógica do robô ultrassônico. . . . .	49

Figura 29 – Distribuição das respostas para as perguntas relacionadas aos componentes e lógica do robô seguidor de linha. . . . .	51
Figura 30 – Distribuição das respostas para as perguntas 1 e 2. . . . .	52
Figura 31 – Distribuição das respostas para as perguntas 3 e 4. . . . .	53
Figura 32 – Distribuição das respostas para a pergunta 5. . . . .	54

## **LISTA DE QUADROS**

Quadro 1 – Quadro para comparação de trabalhos. . . . .	30
---	----

## LISTA DE CÓDIGOS-FONTE

Código-fonte 1 – Código para controle do robô ultrassônico. . . . .	58
Código-fonte 2 – Código base de controle do robô seguidor de linha. . . . .	61

## SUMÁRIO

1	<b>INTRODUÇÃO</b>	13
1.1	<b>Motivação</b>	14
1.2	<b>Objetivos</b>	14
1.2.1	<i>Objetivo Geral</i>	14
1.2.2	<i>Objetivos Específicos</i>	14
1.3	<b>Organização</b>	15
2	<b>FUNDAMENTAÇÃO TEÓRICA</b>	16
2.1	<b>Plano de ensino</b>	16
2.2	<b>Robótica educacional</b>	17
2.2.1	<i>A aplicação da RE na educação</i>	17
2.3	<b>EPRA</b>	18
2.4	<b>Arduino</b>	18
2.4.1	<i>Pinos de alimentação</i>	19
2.4.2	<i>Pinos digitais</i>	20
2.4.3	<i>Pinos analógicos</i>	21
2.4.4	<i>IDE Arduino</i>	21
2.5	<b>Tinkercad</b>	21
2.6	<b>Robô ultrassônico</b>	24
2.7	<b>Robô seguidor de linha</b>	25
3	<b>TRABALHOS RELACIONADOS</b>	28
3.1	<b>Robótica Educacional: Desenvolvimento de Atividades Práticas Utilizando Kits de LEGO Mindstorms EV3 para o Projeto EPRA</b>	28
3.2	<b>Aplicação de Robótica Computacional no Ensino Fundamental em Crateús</b>	28
3.3	<b>Robótica nas aulas de matemática do ensino médio: uma proposta educacional e de baixo custo</b>	29
3.4	<b>Comparação de Trabalhos Relacionados</b>	30
4	<b>METODOLOGIA</b>	31
4.1	<b>Histórico de cursos do projeto EPRA</b>	31
4.1.1	<i>Ano 2023</i>	31
4.1.2	<i>Ano 2024</i>	33

4.2	<b>Plano de curso: Do Básico ao Robô Seguidor de Linha</b> . . . . .	34
4.2.1	<i>Parte 1: Lógica de programação em blocos no Tinkercad</i> . . . . .	35
4.2.2	<i>Parte 2: lógica de programação em C no Tinkercad (Transição de blocos para sintaxe C)</i> . . . . .	38
4.2.3	<i>Parte 3: Lógica de Programação em C com o Kit de Arduino Mega e Chassi 4WD</i> . . . . .	39
4.2.4	<i>Parte 4: Projeto Seguidor de Linha</i> . . . . .	42
4.3	<b>Aplicação da proposta e material de apoio</b> . . . . .	43
4.4	<b>Método de avaliação</b> . . . . .	44
5	<b>ANÁLISE DOS RESULTADOS</b> . . . . .	45
5.1	<b>Análise sobre o uso da plataforma Tinkercad</b> . . . . .	45
5.2	<b>Programação em C</b> . . . . .	47
5.3	<b>Programação do robô ultrassônico</b> . . . . .	48
5.4	<b>Programação do robô seguidor de linha</b> . . . . .	50
5.5	<b>Análise geral</b> . . . . .	52
6	<b>CONCLUSÕES E TRABALHOS FUTUROS</b> . . . . .	55
	<b>REFERÊNCIAS</b> . . . . .	56
	<b>APÊNDICE A –CÓDIGO BÁSICO DE CONTROLE DE UM ROBÔ ULTRASSÔNICO</b> . . . . .	58
	<b>APÊNDICE B –CÓDIGO BASE DE CONTROLE DE UM ROBÔ SEGUIDOR DE LINHA</b> . . . . .	61
	<b>APÊNDICE C –QUESTIONÁRIO PARA AVALIAÇÃO DO PLANO DIRECIONADO AOS ALUNOS</b> . . . . .	64

## 1 INTRODUÇÃO

A constante evolução da tecnologia no século XXI tem reconfigurado os paradigmas educacionais. Em um mundo quase que totalmente conectado, as instituições de ensino são desafiadas a ir além de métodos tradicionais, integrando ferramentas tecnológicas que dialoguem com a realidade dos estudantes. As tecnologias digitais não são meros acessórios, mas elementos capazes de dinamizar o processo de ensino e aprendizagem, tornando-o mais interativo, motivador e alinhado às demandas de uma sociedade em constante transformação (Souza, 2021).

Nesse sentido, a robótica educacional (RE), emerge no Brasil como uma ferramenta estratégica no contexto das demandas tecnológicas do século XXI, capacitando estudantes do ensino superior e médio através do desenvolvimento de habilidades como raciocínio lógico e criatividade. Para isso, a abordagem prática se torna essencial no fortalecimento da aprendizagem teórica e despertando no aluno o interesse pelo conhecimento (Wiltgen, 2022).

A RE surgiu na década de 1960, idealizada por Seymour Papert, que desenvolveu a filosofia de aprendizagem Construcionista, que buscava otimizar o aprendizado dos alunos através da priorização de aulas práticas. Firmado nessa filosofia, em 1967, Papert desenvolveu a linguagem LOGO, com o objetivo de ampliar o estudo da matemática, além de mudar nos alunos sua forma de pensar, criando neles a capacidade de transferir “hábitos da vida real para a construção de teorias científicas” (Santos, 2020).

Não necessariamente derivado das ideias de Papert, mas que ainda assim é capaz de desenvolver um ambiente similar com o que foi desenvolvido pela linguagem LOGO, surge a plataforma de código aberto Arduino (Santos, 2020). Com uma comunidade global, a plataforma Arduino dispõe de uma imensa gama de possibilidades, sendo capaz de unir hardware e software em diversos tipos de projetos, inclusive alguns destinados a educação (Arduino, 2018).

O projeto de extensão universitária EPRA (Ensino de Programação e Robótica com Arduino), da Universidade Federal do Ceará (Campus Russas) destaca-se por promover capacitação prática aos alunos de ensino superior e médio da cidade de Russas no Ceará. O projeto EPRA dispõe de kits LEGO® *Mindstorms* EV3 e Arduino Mega, desenvolvendo em seus cursos desde os fundamentos da linguagem de programação C, até a programação e montagem de projetos utilizando os kits LEGO® ou Arduino (EPRA, 2025).

## 1.1 Motivação

Ao ingressar no ensino superior, principalmente em cursos nas áreas de exatas, os alunos sentem dificuldade em aprender o que ali está sendo ensinado, ocasionando em reprovação e evasão escolar logo no início do curso. Muito dessa dificuldade vem do ensino médio, onde os alunos deveriam aprender o básico, mas devido à falta de estrutura, a maioria dos alunos não aprendem o suficiente para estarem preparados para o ensino superior (Pereira, 2019).

Contudo, a elaboração de um plano de ensino bem estruturado, com aulas praticas e dinâmicas que proporcionem uma transição gradual do uso da programação em blocos para a programação de robôs seguidores de linha, além poder contribuir com a redução dessa evasão, pode transformar o estudo da programação e robótica em uma experiência motivadora para os alunos. Além disso, impulsiona a educação como um todo e prepara os alunos para os desafios do mundo digital.

## 1.2 Objetivos

Na presente seção, serão apresentados os objetivos a serem alcançados neste trabalho, começando pelo objetivo geral e passando para os mais específicos.

### 1.2.1 *Objetivo Geral*

Elaborar um plano de ensino para um curso de programação e robótica focado em aulas práticas, utilizando as plataformas Arduino e Tinkercad, estruturado para promover uma transição gradual da programação em blocos para a sintaxe da linguagem C aplicada a robôs ultrassônicos e seguidores de linha. A proposta é destinada a alunos de ensino superior atendidos pelo projeto EPRA, aplicada em um curso ofertado entre novembro e dezembro de 2025.

### 1.2.2 *Objetivos Específicos*

- Planejar o cronograma e conteúdo das aulas;
- Preparar o material (slides, códigos e circuitos eletrônicos) que serão utilizados nas aulas;
- Aplicar o plano de ensino em um curso ofertado pelo projeto EPRA;
- Validar o plano através da aplicação de formulários e desafio.

### **1.3 Organização**

O restante deste trabalho, será dividido da seguinte forma: No capítulo 2 será apresentada a fundamentação teórica, onde serão abordados temas relacionados a robótica educacional; O capítulo 3 irá apresentar uma seleção de trabalhos relacionados a este; No capítulo 4 o autor descreve a metodologia adotada para realização deste trabalho; O capítulo 5 apresenta a análise dos resultados obtidos; Por último o capítulo 6 com a conclusão.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados conceitos relacionados ao planejamento educacional, robótica educacional e sua aplicação, além de uma visão geral sobre o projeto EPRA. Após isso, serão apresentadas(os) as plataformas Arduino e Tinkercad; e os robôs ultrassônico e seguidor de linha.

### 2.1 Plano de ensino

O planejamento no âmbito educacional surge como uma ferramenta essencial de organização da ação pedagógica. Segundo Puentes (2023), o planejamento é fundamental para evitar a improvisação e estabelecer um caminho seguro que garanta o sucesso da ação educativa. Sant’anna *et al.* (1998) destaca a importância de relacionar o sistema educacional com o desenvolvimento econômico, social, político e cultural do país, para assim alcançar uma maior coerência no planejamento das atividades. Puentes (2023) apresenta os tipos de planejamento:

- **Planejamento Educacional:** O mais abrangente de todos (realizado em esferas nacionais, estaduais e municipais) que define os objetivos gerais e como eles serão atingidos, visando atender às necessidades da sociedade.
- **Planejamento Curricular:** Este nível é relativo à escola e envolve a tomada de decisões sobre a dinâmica da ação escolar.
- **Planejamento de ensino:** Sendo atribuído ao professor, o plano de ensino irá organizar, estruturar e sistematizar as ações docentes e discentes para garantir que os objetivos de aprendizagem sejam atingidos.

Já Santiago *et al.* (2023) destaca que “frequentemente a academia com seu ensino tradicional, com aulas expositivas e grande conteúdo teórico, não atende às necessidades da indústria”, criando uma lacuna na formação dos estudantes. Para reduzir esse problema, é proposto o uso de projetos práticos estruturados através de uma Sequência Didática (SD), visando simular um ambiente real e fixar os conteúdos de forma prática.

Santiago *et al.* (2023) define uma Sequência Didática (SD), como um conjunto de atividades organizadas para atingir os objetivos educacionais. A elaboração dessa estrutura, virá a reduzir dificuldades como a gestão do tempo durante as aulas.

## 2.2 Robótica educacional

A Robótica educacional (RE) é uma metodologia de ensino que utiliza kits de montagem (físicos e virtuais), sensores, atuadores e softwares de programação. Ela proporciona aos alunos uma aprendizagem multidisciplinar e concede total liberdade criativa durante o desenvolvimento de seus projetos. Seu objetivo é promover a construção ativa do conhecimento, estimulando o pensamento criativo, lógico e analítico, além da capacidade de resolver problemas (Santos, 2020).

Idealizada por Seymour Papert, na década de 1960, quando ele iniciou seus estudos e criou a linguagem LOGO, destinada a alunos iniciantes, com o propósito de fazê-los “refletirem sobre a própria forma de pensar”(Santos, 2020), e otimizar o aprendizado. Papert também desenvolveu a filosofia de aprendizagem Construcionista, que propõe a priorização de experiências práticas em relação às teóricas (Santos, 2020).

### 2.2.1 A aplicação da RE na educação

No contexto do ensino médio, a robótica atua como uma ferramenta essencial na vida do estudante, indo além do simples aprendizado técnico. Segundo Braga (2020), o estudo dessa temática e da codificação nesta fase pode “revelar uma paixão oculta ou iluminar uma possível trajetória profissional”, ensinando habilidades técnicas valorizadas no mercado de trabalho, como a própria programação. Além disso, o autor destaca que, com o auxílio da programação de robôs, os alunos desenvolvem competências de “ordem superior”, como a “resolução de problemas e raciocínio”, incentivando-os a pensar de forma criativa e a trabalhar efetivamente em equipe (Braga, 2020).

No ensino superior, especialmente nas engenharias, um dos maiores desafios é conectar a densa base teórica com a prática profissional. Braga (2020) aponta que a robótica educacional entra nesse cenário como uma “ponte”, permitindo que alunos de graduação visualizem fisicamente conceitos abstratos de Cálculo e Física. Segundo o autor, ao ver as equações funcionando no movimento de um robô, o universitário enxerga sentido no que estuda, o que combate a desmotivação comum nos primeiros anos do curso (Braga, 2020).

A implementação da robótica na graduação frequentemente assume a forma de disciplinas práticas ou projetos de extensão, nos quais os estudantes são desafiados a desenvolver protótipos funcionais. Wiltgen (2022) relata uma experiência bem-sucedida em um curso de

Engenharia Elétrica, onde os alunos projetaram e construíram robôs para competições internas, como labirintos e circuitos externos. Esse tipo de atividade, com cerca de 60–70% de carga prática, permitiu que os estudantes vivenciassem todas as etapas do desenvolvimento de um sistema robótico desde o projeto e a seleção de componentes até a programação, montagem e testes, consolidando conhecimentos de eletrônica, programação, controle e mecânica (Wiltgen, 2022).

Os benefícios observados são significativos, tanto no desempenho acadêmico quanto na formação profissional. Wiltgen (2022) documentou um aumento expressivo no interesse por temas de robótica e automação entre os graduandos, com muitos optando por desenvolver Trabalhos de Conclusão de Curso na área. Dessa forma, a robótica educacional no ensino superior posiciona-se como uma estratégia pedagógica essencial que conecta a formação acadêmica às necessidades da sociedade e da indústria.

### **2.3 EPRA**

O EPRA (Ensino de Programação e Robótica com Arduino), é um projeto de extensão da Universidade Federal do Ceará, campus de Russas, coordenado pelos professores Pablo Soares (criador do projeto) e Reuber Regis. O EPRA atua desde 2018 ofertando curso de programação básica e robótica às escolas de ensino médio da cidade de Russas-CE (EPRA, 2025).

O projeto oferta em média dois cursos ao ano, o primeiro de caráter mais básico e o segundo transitando para assuntos mais avançados. Até 2023 o projeto utilizava apenas kits Arduino Mega para as aulas de robótica, mas devido a dificuldade no aprendizado dos conceitos básicos e pensamento lógico dos alunos, em 2024 os tutores incluíram a utilização dos kits LEGO® *Mindstorms* EV3, que traz uma interface mais intuitiva, facilitando a compreensão dos alunos (EPRA, 2025).

### **2.4 Arduino**

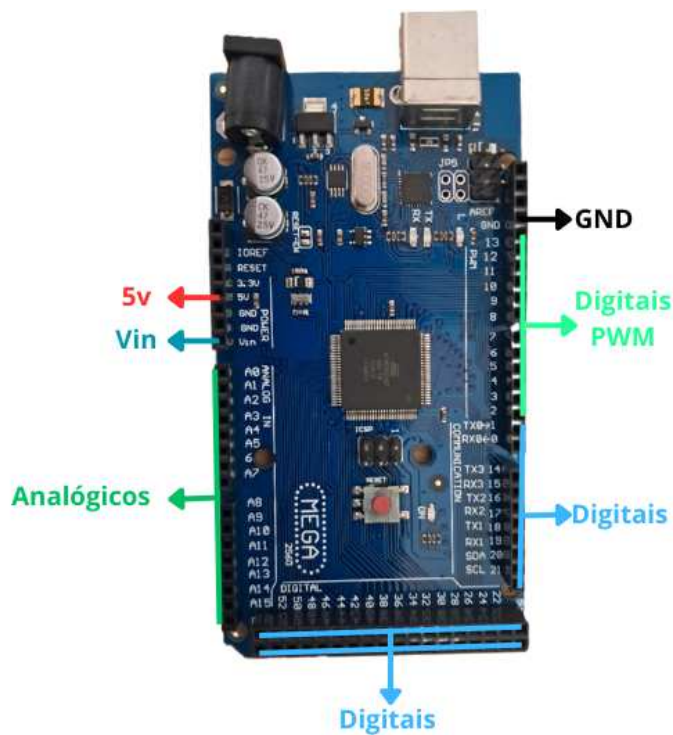
O Arduino é uma tecnologia *open-source* que integra hardware e software, através de placas eletrônicas programáveis. Além de possibilitar o desenvolvimento de diversos tipos de projetos, o Arduino também pode ser aplicado em diversas áreas como robótica, educação e IoT (Internet das Coisas) (Arduino, 2018).

O Arduino democratiza o acesso à tecnologia, graças ao seu baixo custo, natureza

multiplataforma (compatível com Windows, Linux e macOS) e possuir uma comunidade global ativa, que sustenta a plataforma compartilhando conhecimento e inovação (Arduino, 2018).

As placas Arduino, possuem pinos digitais e analógicos, capazes de receber dados de sensores, processá-los e emitir saídas através de atuadores. A Figura 1 exibe uma placa Arduino modelo Mega e a indicação de seus principais pinos.

Figura 1 – Placa Arduino Mega e pinagem



Fonte: Elaborado pelo autor

#### 2.4.1 Pinos de alimentação

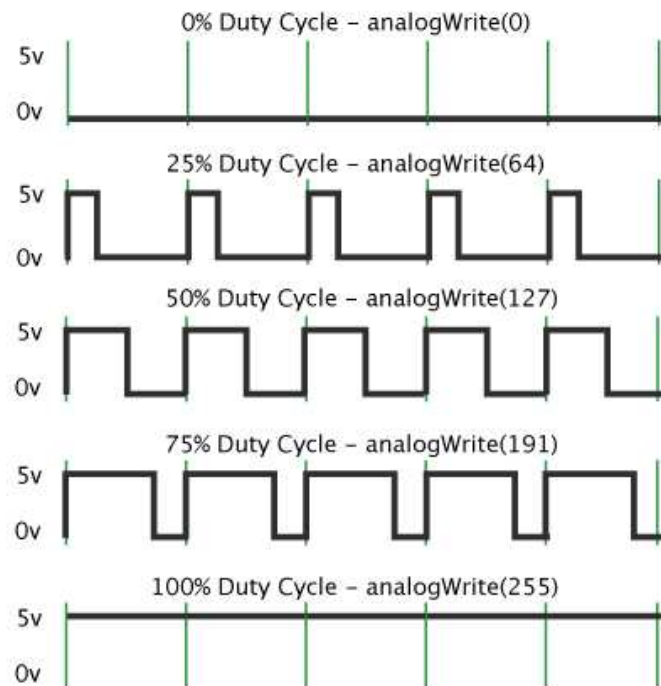
Os pinos 5v, Vin e gnd são responsáveis por fornecer alimentação para os componentes e circuitos conectados ao arduino. O pino 5v fornece 5 volts de tensão; O Vin é conectado a entrada de fonte externa fornecendo a tensão que recebe desta (ex: 9 ou 12 volts), logo permitirá o uso de componentes que exijam mais do que 5v de alimentação; E o gnd (*Ground*) representa o referencial de tensão zero (terra) do circuito (Arduino, 2018).

### 2.4.2 Pinos digitais

Os pinos digitais podem ser programados para funcionar como entradas ou saídas, operando com apenas dois estados lógicos: HIGH (5V) ou LOW (0V). Como Entrada ele poderá ler o estado de um componente externo (ex.: botão pressionado ou solto); E como Saída fornecer energia para controlar um componente (ex.: acender um LED) (Arduino, 2018).

Os pinos digitais PWM possuem a capacidade de gerar sinais de Modulação por Largura de Pulso (Figura 2), simulando uma saída analógica. Isso é essencial para controlar a velocidade de motores ou o brilho de LEDs (Arduino, 2018).

Figura 2 – Modulação de largura de pulso



Fonte: <https://docs.arduino.cc/>

A Modulação por Largura de Pulso é uma técnica que permite controlar dispositivos digitais através da variação do tempo em que um sinal permanece em estado lógico alto (HIGH) durante um ciclo periódico fixo. Em microcontroladores como o Arduino, o PWM não gera uma saída analógica verdadeira, mas sim simula-a através da rápida alternância entre os estados HIGH (5V) e LOW (0V) em um pino digital (Arduino, 2018).

### 2.4.3 Pinos analógicos

Quanto aos pinos analógicos, sua função principal é ler valores de tensão contínua em uma faixa de 0V a 5V, convertendo-os em um valor entre 0 e 1023. Esta característica é indispensável para utilizar sensores analógicos, como sensores de luminosidade (LDR) ou sensores de distância (Arduino, 2018).

### 2.4.4 IDE Arduino

Para programação e *upload* do código na placa, a plataforma disponibiliza a IDE Arduino que funciona utilizando a linguagem C++ simplificada (Arduino, 2018). A Figura 3 exibe a IDE Arduino.

Figura 3 – Interface de programação Arduino



Fonte: IDE Arduino

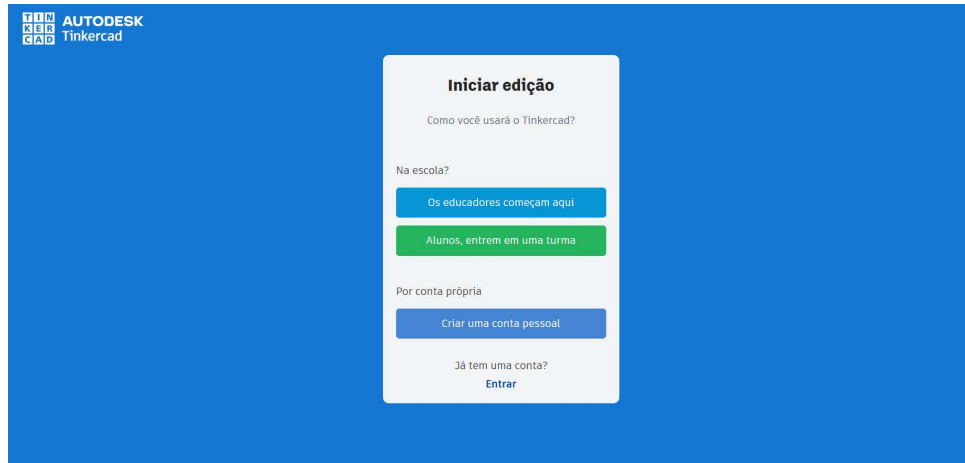
## 2.5 Tinkercad

A plataforma Tinkercad consiste em uma ferramenta online gratuita, desenvolvida pela Autodesk, que permite a criação de modelos 3D e a simulação de circuitos eletrônicos com Arduino.

Segundo Nascimento (2024), o Tinkercad se destaca no contexto da Robótica Educacional por oferecer um ambiente de fácil acesso e usabilidade, e permite a experimentação sem riscos, como a danificação dos componentes eletrônicos. A plataforma permite que os

professores gerenciem suas turmas através de contas específicas para o ensino. A Figura 4 mostra a pagina de inscrição da plataforma com a opção de educadores.

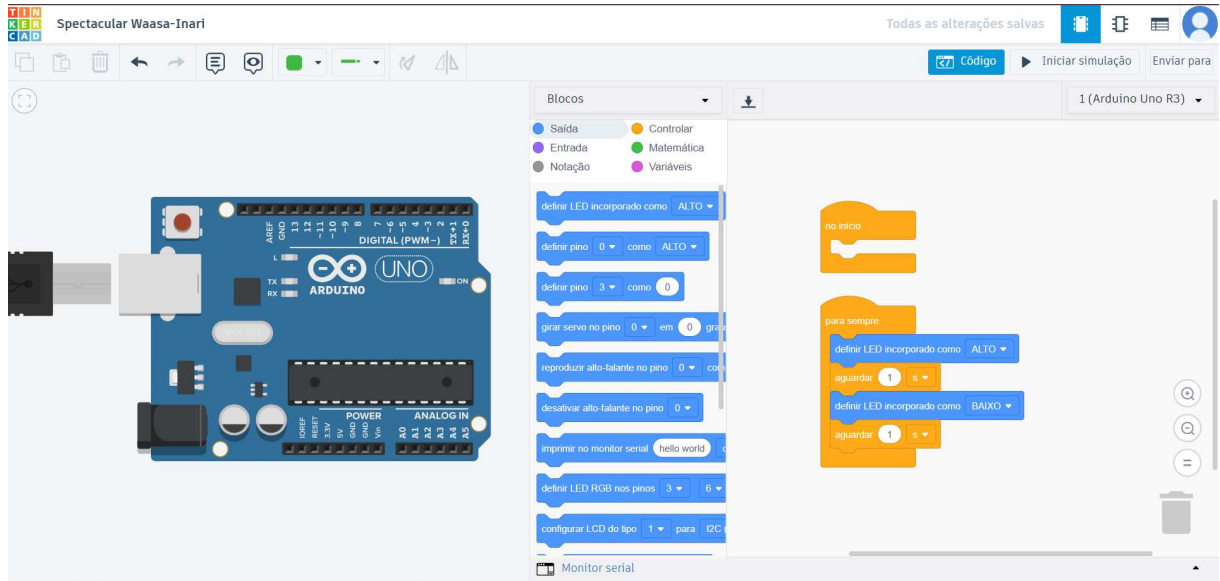
Figura 4 – Página de inscrição do Tinkercad



Fonte: <https://www.tinkercad.com/>

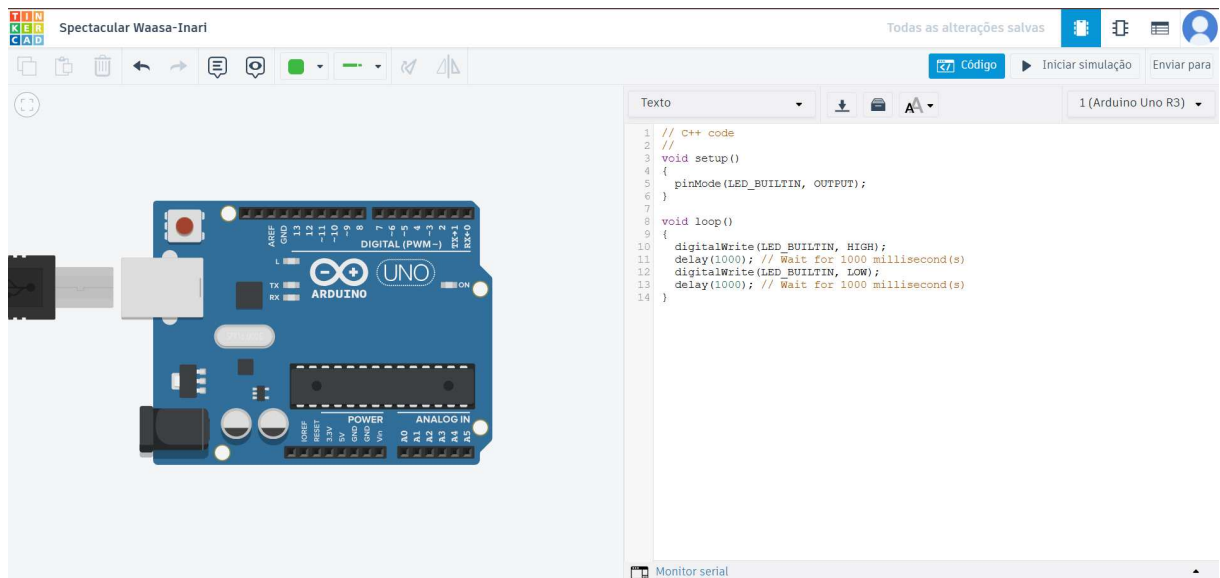
A plataforma integra um editor de código que suporta tanto a montagem de algoritmos com blocos quanto a programação em linguagem C++, facilitando o aprendizado de conceitos básicos da programação, antes mesmo de ser ensinado a sintaxe textual da linguagem e se tornando uma alternativa de baixo custo à plataforma LEGO® *Mindstorms* EV3. A Figura 5 exibe a interface de programação em blocos, enquanto a Figura 6 em texto.

Figura 5 – Interface de programação em blocos



Fonte: <https://www.tinkercad.com/>

Figura 6 – Interface de programação em texto



Fonte: <https://www.tinkercad.com/>

Conforme aponta Nascimento (2024), o uso desse simulador em práticas educativas, potencializa a autonomia do estudante, permitindo que ele visualize o funcionamento lógico e eletrônico do projeto antes da montagem física.

## 2.6 Robô ultrassônico

O robô ultrassônico é um tipo de robô autônomo que utiliza sensores ultrassônicos (Figura 7) para percepção do ambiente, especialmente para detecção e medição de distâncias de objetos em seu caminho.

Figura 7 – Sensor ultrassônico



Fonte: <https://www.sta-eletronica.com.br/>

Diferentemente do robô seguidor de linha que segue uma trajetória predeterminada, o robô ultrassônico possui capacidade de navegação reativa, tomando decisões em tempo real com base nas informações coletadas por seus sensores (Eletronica, 2024).

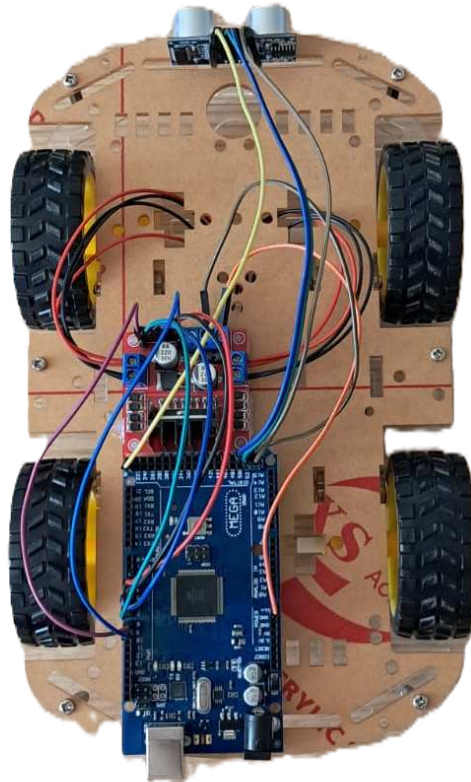
O sensor HC-SR04, amplamente utilizado em projetos educacionais e de prototipagem, opera com base no princípio de medição por tempo de voo. Seu funcionamento pode ser descrito em três etapas sequenciais:

1. **Emissão:** O transdutor do sensor (conectado ao pino Trigger) emite um pulso de ondas ultrassônicas em uma frequência de aproximadamente 40 kHz, inaudível ao ser humano.
2. **Propagação e Reflexão:** As ondas sonoras se propagam pelo ar. Se encontrarem um objeto ou obstáculo em seu caminho, são refletidas de volta para o módulo sensor.
3. **Recepção e Cálculo:** O transdutor atua como microfone para receber o eco (pino Echo). O circuito do sensor mede o intervalo de tempo entre a emissão do pulso e a recepção do seu eco. Conhecendo a velocidade do som no ar, a distância até o objeto é calculada (Eletronica, 2024).

Dessa forma, o robô consegue detectar possíveis impactos futuros e mudar de direção

antes que aconteça. A Figura 8 exibe um robô ultrassônico montado pela equipe do projeto EPRA.

Figura 8 – Robô ultrassônico



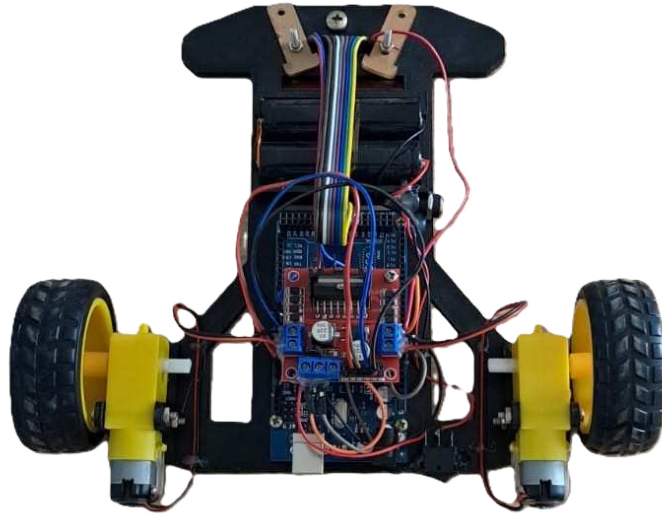
Fonte: Arquivos do projeto EPRA

## 2.7 Robô seguidor de linha

Robôs seguidores de linha são considerados robôs móveis AVG (Veículos Auto Guiados), capazes de se movimentar de forma autônoma em circuitos pré-montados conforme mostra a Figura 9 (Virmond, 2023).



Figura 10 – Robô seguidor de linha e indicação de componentes



Fonte: Arquivos do projeto EPRA

### 3 TRABALHOS RELACIONADOS

Esta seção irá apresentar uma seleção de trabalhos relacionados a esta pesquisa, seguido pela comparação.

#### 3.1 Robótica Educacional: Desenvolvimento de Atividades Práticas Utilizando *Kits* de LEGO *Mindstorms* EV3 para o Projeto EPRA

Nogueira (2024) traz como objetivo principal, o desenvolvimento de um plano de atividades, para o ensino de robótica utilizando *kits* de LEGO® destinado a alunos de ensino médio da cidade de Russas, Ceará. O autor fez uma pesquisa em algumas escolas da cidade, para saber se a robótica era um assunto presente entre os assuntos estudados, e infelizmente constatou-se ausência de disciplinas ou projetos contínuos de robótica. Adicionalmente o autor desenvolveu um plano de atividades, com um total de 6 aulas (12 horas), divididas em:

- Introdução a Robótica, ao LEGO® e apresentação de robôs.
- Componentes do LEGO® *Mindstorms* EV3 - Parte 1 (Blocos e motores).
- Componentes do LEGO® *Mindstorms* EV3 - Parte 2 (sensores).
- Montagem de um robô seguidor de linha
- Estruturas de repetição e escolha com exemplos práticos.
- Desafio.

A aplicação e validação foram feitas através de um projeto de extensão da Universidade Federal do Ceará - Campus de Russas, chamado Ensino de Programação e Robótica com Arduino (EPRA). O projeto aplicou o plano em um curso com 32 alunos no total e avaliou-os com um desafio final. Os resultados foram colhidos através do desafio final e questionários. No desafio final, 70% das equipes completaram todas as atividades, sendo vista como principal dificuldade o uso combinado de estruturas de repetição e condicionais. Nos questionários cerca de 81% das respostas indicaram o uso dos *kits* LEGO® como “confortável”, levando em consideração sua facilidade de montagem e programação. O plano mostrou ser eficaz para introduzir o estudo da robótica no ensino médio.

#### 3.2 Aplicação de Robótica Computacional no Ensino Fundamental em Crateús

Leitão (2024) investiga o impacto da robótica educacional no ensino fundamental da rede pública da cidade de Crateús-CE, com o objetivo de reduzir a evasão escolar e despertar o

interesse dos alunos por tecnologia. Para isso, ele propôs a realização de uma oficina de robótica, dividida nos seguintes módulos:

- **Módulo 1:** Conceitos de robótica, eletrônica e pensamento computacional.
- **Módulo 2:** Uso da plataforma *Autodesk Tinkercad* para simulação de circuitos e programação em blocos (adaptado para C++ devido a problemas de infraestrutura).
- **Módulo 3:** Desenvolvimento de protótipos com Arduino em grupos, apresentados em feira de ciências.

Os resultados foram colhidos através de questionários aplicados ao longo da oficina, foi apontado 90,9% de satisfação por parte dos alunos onde 45,5% manifestaram forte interesse em seguir carreira na área. Foi concluído que a abordagem prática com o uso de Arduino, foi crucial para despertar o interesse dos alunos, além de aumentar o engajamento dos alunos e destruir barreiras abstratas no aprendizado.

### 3.3 Robótica nas aulas de matemática do ensino médio: uma proposta educacional e de baixo custo

Junior (2017) propõe a implementação de baixo custo da robótica educacional em aulas de matemática do ensino médio, para promover interdisciplinaridade, autonomia intelectual e engajamento dos alunos. O projeto foi dividido em três etapas. A primeira etapa foi a análise comparativa dos *kits* robóticos que poderiam ser usados nas etapas seguintes. Os *kits* avaliados foram:

- **LEGO® Mindstorms:** Muito usado na educação básica por sua facilidade de manuseio e programação intuitiva com blocos. Porém devido ao seu alto custo se tornou uma opção inviável.
- **Boe-bot:** Programação em linguagem basic, que a tornou uma boa opção por ser simples de ensinar. Mas seu material didático vem todo em inglês, podendo dificultar a compreensão do seu manuseio.
- **Arduino:** Foi a opção mais viável encontrada, devido seu baixo custo e modularidade.

Em seguida o autor desenvolveu um tutorial pedagógico, estruturado em duas atividades aplicadas em 2 aulas com 45 minutos cada:

- **Atividade 1:** Introdução à robótica e revisão de conceitos teóricos (funções de 1º grau);
- **Atividade 2:** Aplicação prática com programação do robô para resolver problemas matemáticos (ex: relações espaço-tempo, gráficos).

Como fundamentação pedagógica foram utilizados os modelos de trabalho independente (Libaneo, 2004) e cenários de investigação (Skovsmose, 2001; Skovsmose, 2008). Durante a implementação das aulas, foi obtido um alto engajamento dos alunos onde 72% relataram maior facilidade na visualização de conceitos abstratos.

### 3.4 Comparação de Trabalhos Relacionados

O Quadro 1 exhibe as semelhanças e diferenças dos trabalhos relacionados e o trabalho realizado e descrito neste documento.

Quadro 1 – Quadro para comparação de trabalhos.

Trabalho	Semelhanças	Diferenças
Nogueira (2024)	Criação de um plano de ensino de robótica educacional e aplicação num curso ofertado pelo projeto EPRA.	Além da diferença de carga horária onde Nogueira (2024) propõe um plano com 6 aulas totalizando 12 horas (2 horas/aula), ele utiliza os kits LEGO® <i>Mindstorms</i> como material de apoio em todas as aulas propostas e tem como público alvo, alunos de ensino médio. Enquanto o plano proposto no presente trabalho propõe 12 aulas totalizando 24 horas (2 horas/aula), utiliza a plataforma Arduino como material de apoio e tem como público alvo alunos de ensino superior.
Leitão (2024)	Aplicação da robótica educacional, utilização da plataforma Tinkercad e kits Arduino como material de apoio.	Leitão (2024) busca atender alunos de ensino fundamental, além de propor uma metodologia em forma de oficina. Enquanto o presente autor busca atender alunos de ensino superior e propõe um plano de ensino.
Junior (2017)	Criação de um plano de ensino de robótica e utilização de kits Arduino.	Junior (2017) desenvolve apenas duas aulas de 45 minutos cada, além de ter como público alvo alunos de ensino médio. Enquanto o presente autor propõe um plano com 12 aulas e tem como público alvo alunos de ensino superior

Fonte: Elaborado pelo autor.

## 4 METODOLOGIA

Este capítulo irá apresentar a metodologia adotada para a realização deste trabalho, onde será realizada uma análise do histórico recente de cursos promovidos pelo projeto EPRA, seguido pela proposta do plano de ensino a ser desenvolvido e finalizando com o método de avaliação.

### 4.1 Histórico de cursos do projeto EPRA

O presente autor integra a equipe do projeto desde o primeiro semestre de 2023, atuando como tutor nas áreas de programação e robótica. Esta análise tem como ponto de partida esse período, considerando as experiências vivenciadas até a presente data.

#### 4.1.1 Ano 2023

No primeiro semestre de 2023 o projeto iniciou o ano com um curso de programação básica em linguagem C, neste curso não foram apresentados conceitos de robótica ou Arduino, somente os fundamentos da linguagem utilizando a ide Code::Blocks. A equipe utilizou como material base de ensino, uma apostila desenvolvida no ano de 2020 pelos tutores do projeto EPRA presentes na época.

A apostila apresentava desde conceitos básicos como variáveis, operadores, estruturas de repetição, condicionais e funções; Até conceitos mais avançados como vetores, matrizes, strings e tipos estruturados (struct, enum e typedef). Porém antes do início do curso (2023.1) a equipe percebeu a necessidade de reduzir no plano, grande parte dos conceitos mais avançados devido ao fator tempo e ao perceber que esses elementos não seriam utilizados mais para frente durante o curso de robótica e arduino. A Figura 11 mostra o plano de ensino utilizado.

Figura 11 – Roteiro de aulas do curso 2023.1

Aula	Capítulo da apostila
1	1: apresentação e conceitos básicos
2	2: estrutura linguagem c, var, scan, print
3	3: operadores e condicionais
4	4: Prática da aula 3
5	5: Estruturas de repetição
6	6: Prática aula 5
7	7: vetores e strings
8	8: Prática aula 7
9	9: Funções
10	10: Prática aula 9
11	11: Revisão para avaliação 1
12	10: Avaliação módulo 1

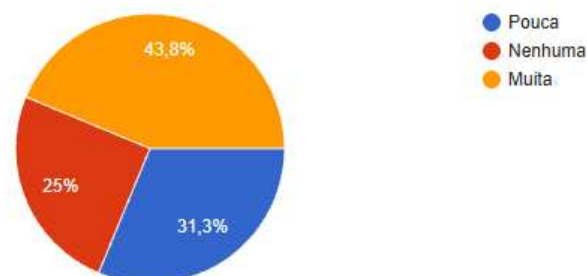
Fonte: Arquivos do projeto EPRA

Ao final do curso, a equipe de tutores aplicou um formulário aos alunos com o objetivo de coletar informações relacionadas a satisfação com o aprendizado adquirido durante o curso. O gráfico mostrado na Figura 12, mostra o nível de dificuldade que alguns alunos apresentaram no decorrer do curso.

Figura 12 – Nível de dificuldade

Você está encontrando dificuldades com a programação no curso?

16 respostas



Fonte: Arquivos do projeto EPRA

Ao perceber a dificuldade dos alunos, a equipe do projeto EPRA realizou modificações na metodologia de ensino. Foi eliminado o uso da ide Code::Blocks, e adotada a

plataforma Tinkercad, que além de possibilitar o ensino dos fundamentos da linguagem C, é capaz de simular circuitos eletrônicos em uma interface simples e intuitiva sem necessidade de componentes físicos.

No segundo semestre de 2023, o projeto ofertou um curso focado no ensino da robótica e Arduino. A Figura 13 exibe o plano de ensino adotado no curso.

Figura 13 – Roteiro de aulas do curso 2023.2

Aula	Dia	Conteúdo
1	18-19/09	Introdução ao arduino
2	20-21/09	O software e linguagem c
3	25-26/09	Funções no arduino
4	27-28/09	Entradas e saídas
5	02-03/10	Componentes: Protoboard, led, e resistores/ jumpers
6	04-05/10	Pratica
7	09-10/10	componentes: Botões, buzzer e potenciômetros
8	11-17/10	Pratica
9	16-19/10	Componentes: Motores(servo, passo, CC), sensor ultrasonico.
10	18-24/10	Pratica
11	23-26/10	Apresentação do robô
12	25-31/10	Programação e prática com o robô
13	30-07/10-11	Pratica
14	01-09/11	pratica
15	06-14/11	Competição (pratica)
16	08-16/11	Competição (Avaliação)

Fonte: Arquivos do projeto EPRA

Este curso iniciou com o uso do Tinkercad (aulas 1 a 4), onde foi apresentada a placa Arduino Mega como programá-la. Das aulas 5 a 10, foram apresentados alguns componentes eletrônicos além de desafios práticos com eles. Na ultima parte do curso (aulas 11 a 16) foi iniciada a preparação para o desafio final que ocorreu na aula 16, onde os alunos foram divididos em equipes e desafiados a montar e programar os Kits Chassi 4wd e integrar a ele o Arduino mega, resultando num robô ultrassônico exibido na Figura 8.

Para a competição final a equipe de tutores preparou dois desafios, em que os alunos deveriam, em um tempo limite, ajustar a programação dos robôs, para fazê-los desviar de alguns obstáculos.

#### 4.1.2 Ano 2024

Em 2024, a equipe do projeto realizou novamente modificações na metodologia do curso, dividindo-o em três módulos conforme mostra a Figura 14.

Figura 14 – Roteiro de aulas do curso 2024

Aula	Dia	Conteúdo
<b>Modulo lego</b>		
1	8,9/04	Introdução a Robótica, ao Lego e apresentação de robôs
2	10,11/04	Componentes do Lego Mindstorms EV3 - Parte 1 (Bloco e motores)
3	15,16/04	Componentes do Lego Mindstorms EV3 - Parte 2 (Sensores)
4	17,18/04	Montagem de um robô seguidor de linha com o Lego Mindstorms EV3
5	22,23/04	Estruturas de repetição e escolha com exemplos práticos
6	24,25/04	Desafio.
<b>Modulo linguagem C</b>		
7	29,30/04	Introdução a linguagem C, Tinkercad.
8	6,2/05	Estrutura linguagem C - Var, funções (somente chamadas), bibliotecas...
9	8,7/05	Pratica utilizando Tinkercad
10	13,9/05	Operadores e condicionais
11	15,14/05	Pratica utilizando Tinkercad
12	20,16/05	Estruturas de repetição
13	22,21/05	Pratica utilizando Tinkercad
14	27,23/05	Funções (Criação)
15	29,28/05	Pratica utilizando Tinkercad
<b>Modulo arduino fisico</b>		
16	3,4/06	Introdução ao arduino, ide arduino, entradas e saídas no arduino (pinos digitais, analógicos, ~PWM)
17	5,6/06	Componentes: Protoboard, led, e resistores/ jumpers, buzzers e botão
18	10,11/06	Componentes: Motores(Servo, passo,CC),Ponte h, sensor ultrasonico.
19	12,13/06	Pratica
20	17,18/06	Pratica
21	19,20/06	Pratica e desafio final por turma
		Eliminatórias por turma
22	26/06	Desafio entre turmas

Fonte: Arquivos do projeto EPRA

Foi adicionado ao curso, o uso dos kits LEGO<sup>®</sup> *Mindstorms*. A adição foi idealizada por Nogueira (2024), que desenvolveu seis aulas com o objetivo de aprimorar o raciocínio lógico dos alunos e prepará-los para o próximo módulo. O segundo módulo do curso foi focado no ensino dos fundamentos da programação na linguagem C, abordando conceitos como uso de variáveis, funções, condicionais e estruturas de repetição.

No terceiro e último módulo, foi iniciado o uso da placa Arduino Mega juntamente com a ide Arduino. Os tutores apresentaram aos alunos alguns componentes eletrônicos e prepararam-os para o desafio final, similar ao que ocorreu no curso anterior (2023.2).

#### 4.2 Plano de curso: Do Básico ao Robô Seguidor de Linha

O plano de ensino proposto neste trabalho é focado no uso da plataforma Arduino, dispensando a utilização dos kits LEGO<sup>®</sup>. A intenção é de minimizar o custo do material e agilizar a adaptação dos alunos à programação em linguagem e a robótica com Arduino, de modo que, ao final do curso sejam capazes de programar um robô seguidor de linha. O plano foi

dividido em quatro partes conforme a Figura 15 exibe.

Figura 15 – Roteiro de aulas do curso 2025.2

AULA	CONTEÚDO
<b>Lógica de programação em blocos no Tinkercad</b>	
1	INTRODUÇÃO AO CURSO
2	LAÇOS DE REPETIÇÃO
3	CONDICIONAIS
<b>Lógica de programação em C no Tinkercad (Transição de blocos para sintaxe C)</b>	
4	INTRODUÇÃO A LINGUAGEM C
5	LAÇOS DE REPETIÇÃO EM C
6	CONDICIONAIS E FUNÇÕES
<b>Lógica de Programação em C com o Kit de Arduino Mega e Chassi 4WD</b>	
7	INTRODUÇÃO AO ROBÔ ULTRASSÔNICO, MOTORES DC E PONTE H.
8	SENSOR ULTRASSÔNICO E SUA APLICAÇÃO AO ROBÔ
9	DESAFIO COM OS ROBÔS ULTRASSÔNICOS
<b>Projeto Seguidor de Linha</b>	
10	INTRODUÇÃO AO ROBÔ SEGUIDOR DE LINHA.
11	DESAFIO INICIO
12	DESAFIO FINAL E ENTREGA DOS CERTIFICADOS

Fonte: Elaborado pelo autor

#### **4.2.1 Parte 1: Lógica de programação em blocos no Tinkercad**

Para não perder totalmente os benefícios didáticos da plataforma LEGO®, as três aulas iniciais adotarão a programação em blocos disponibilizada pela plataforma Tinkercad. Essa funcionalidade, irá permitir que os alunos aprendam os conceitos básicos da programação e robótica de forma intuitiva e simples.

Na primeira aula foi apresentado o projeto de extensão EPRA e a estrutura do curso. São introduzidos de forma abrangente os componentes básicos de um robô (sensores, atuadores e controlador), com ênfase nas placas Arduino. Ao final da aula foi apresentada a plataforma Tinkercad, e os alunos foram convidados a se inscrever e entrar na classe através de um código que a própria plataforma disponibiliza (Figura 16).

Figura 16 – Instruções para inscrição na classe

Entrar EPRA 2025.2 com um link ou insira este código de aula:

ED3 MEB QPA

Ao acrescentar alunos a esta aula, concordo que tenho autoridade em nome da minha escola ou distrito para autorizar a coleta de informações pessoais para meus alunos e dar permissão para que meus alunos usem o Tinkercad de acordo com o [Aviso direto](#). Saiba mais sobre a abordagem do Tinkercad para [privacidade e segurança de dados de crianças](#).

Copiar linkCopiar código

**Instruções para o aluno**

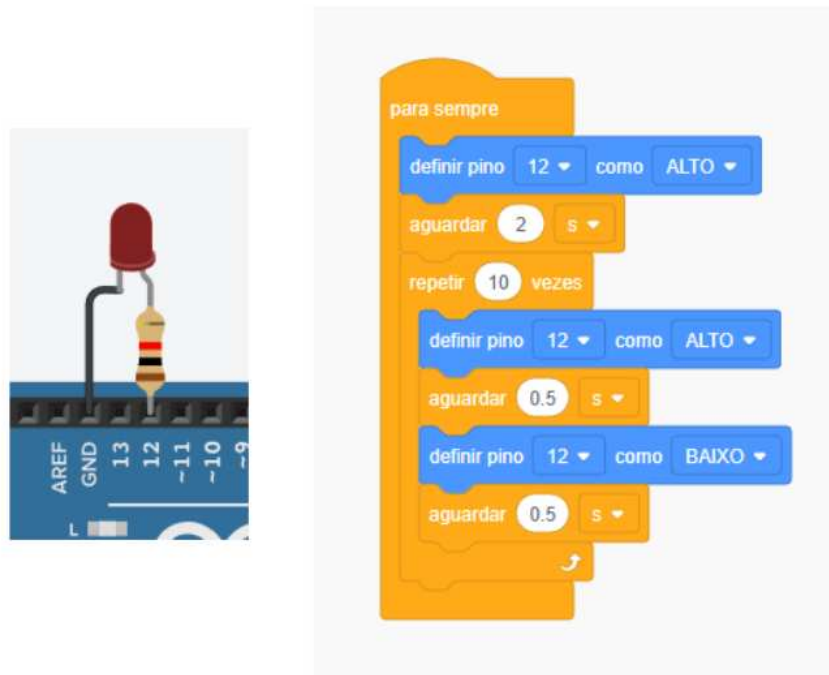
Link da aula:  
1. Vá para sua aula com este link: <https://www.tinkercad.com/joinclass/ED3MEBQPA>  
2. Digite o código de login atribuído pelo professor.

Código de classe:  
1. Vá para <https://www.tinkercad.com/joinclass>  
2. Insira o código de classe: **ED3MEBQPA**  
3. Digite o código de login atribuído pelo professor.

Fonte: <https://www.tinkercad.com/>

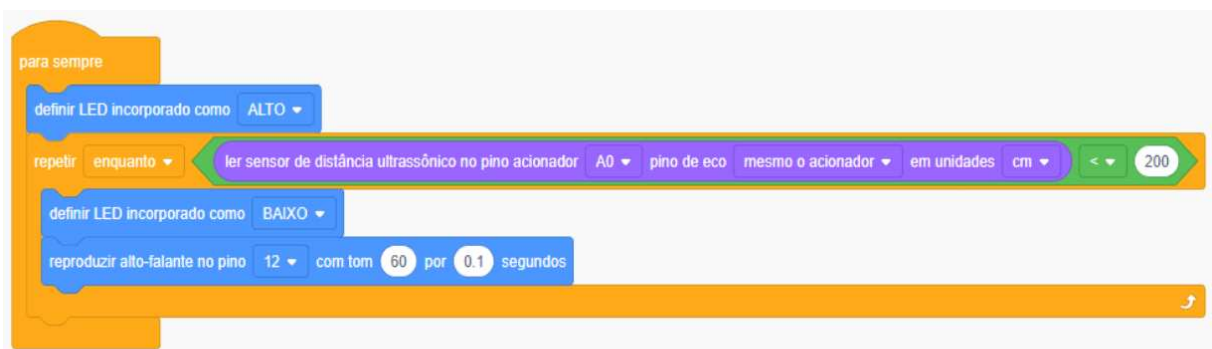
A segunda aula aborda o conceito de estruturas de repetição na programação, utilizando exemplos práticos no Tinkercad, como o controle de um LED (Figura 17) e sensores de estacionamento (Figura 18 e Figura 19), visando ensinar a lógica de programação por meio de atividades interativas.

Figura 17 – Circuito e algoritmo do exemplo: Controle de um LED com arduino



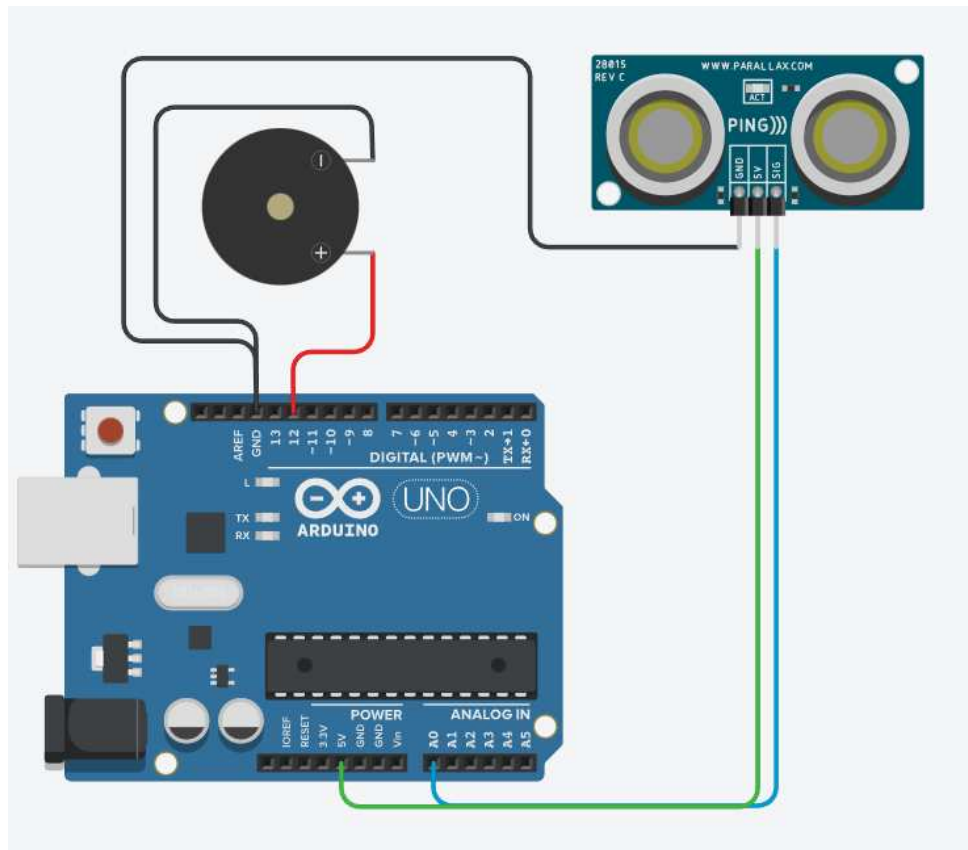
Fonte: <https://www.tinkercad.com/>

Figura 18 – Algoritmo do exemplo: Sensor de estacionamento com arduino



Fonte: <https://www.tinkercad.com/>

Figura 19 – Circuito do exemplo: Sensor de estacionamento com arduino



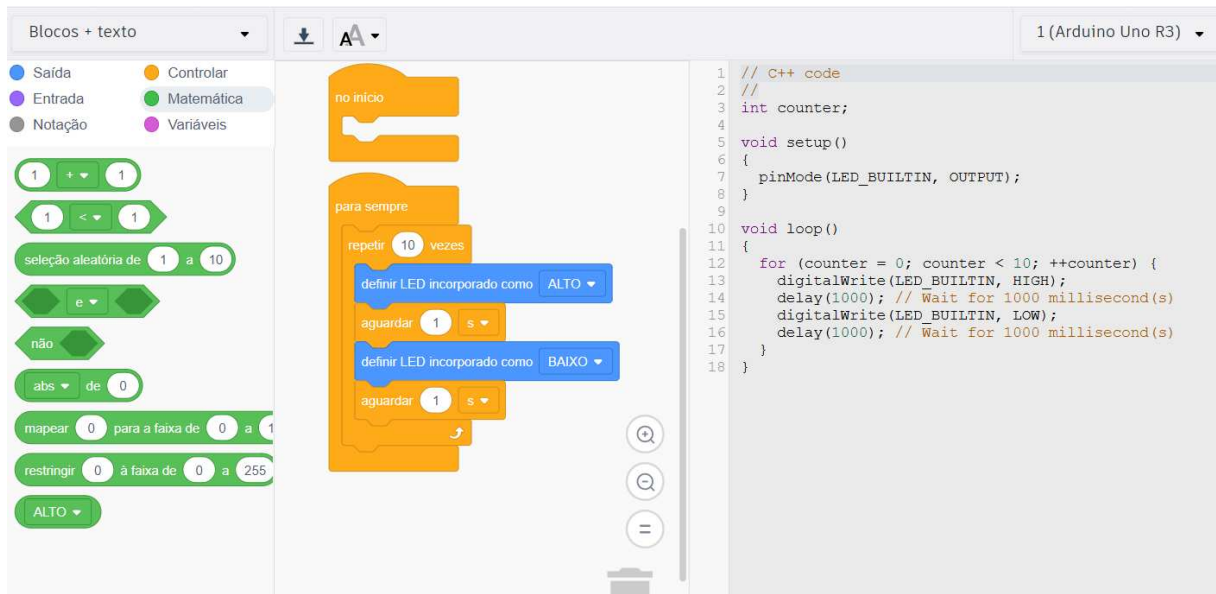
Fonte: <https://www.tinkercad.com/>

A terceira aula introduz o uso de estruturas condicionais na programação, aplicando-as em simulações como a do sensor de estacionamento (Figura 19) no Tinkercad, com exercícios que exigiam que os alunos corrigissem e completassem os algoritmos.

#### 4.2.2 Parte 2: lógica de programação em C no Tinkercad (Transição de blocos para sintaxe C)

Na segunda parte do curso, os alunos começarão a ter contato com a sintaxe da linguagem C. O Tinkercad possui uma função que converte o algoritmo em blocos num código em C, o que irá permitir uma transição suave entre as duas formas de programar. A Figura 20 exhibe um exemplo de um algoritmo feito com blocos em um código em C.

Figura 20 – Conversão blocos para código



Fonte: <https://www.tinkercad.com/>

Na quarta aula foi apresentada a estrutura básica da programação em arduino, mostrando as funções setup e loop e qual a sua funcionalidade; Além de apresentar os principais pinos do arduino (digitais e analógicos). E para fixar, atividades práticas de conversão de algoritmos de blocos para código C e manipulação de componentes como LEDs e buzzer.

Na aula cinco uma revisão e aprofundamento das estruturas de repetição (agora apresentando a sintaxe na linguagem C), com exemplos de incremento, decremento e aplicação em exercícios que envolvem o controle de LEDs por meio de laços de repetição.

Enquanto na aula seis um aprofundamento dos condicionais e sua sintaxe relacionada. Além disso, explorar a definição e utilização de funções em C, destacando benefícios como reutilização de código e modularidade. Finalizando com exercícios práticos com funções para controle de LEDs e sensores ultrassônicos.

### 4.2.3 Parte 3: Lógica de Programação em C com o Kit de Arduino Mega e Chassi 4WD

Na terceira parte, os alunos irão iniciar o contato com o Arduino físico. Isso através de aulas práticas utilizando robôs já montados (com chassi 4wd) para facilitar o aprendizado. Essas três aulas (7 a 9), irão reforçar conceitos básicos como estruturas de repetição e condicionais, que são os conceitos que os alunos mais apresentam dificuldade em aprender.

Na aula sete foi apresentado aos alunos o robô ultrassônico e seus componentes. Nesse momento o foco seria no funcionamento dos motores DC. Ao final da apresentação, os alunos deveriam realizar alguns ajustes num código disponibilizado através do Tinkercad (testando no circuito simulado) e transferir para o robô físico através da IDE Arduino, com o objetivo de testar a sua movimentação.

A aula oito aborda o funcionamento e a programação do sensor ultrassônico para detecção de obstáculos. Aproveitando o código desenvolvido até a aula 7, como atividade os alunos deveriam implementar a lógica de programação do sensor ao robô. O Apêndice A mostra um código básico para o controle completo de um robô ultrassônico.

Como finalização da terceira parte, a aula nove propõe um desafio prático em que os alunos devem ajustar seu código do robô ultrassônico para que ele consiga entrar e sair de uma garagem simulada (Figura 21) sem colidir, aplicando conceitos aprendidos nas aulas anteriores.

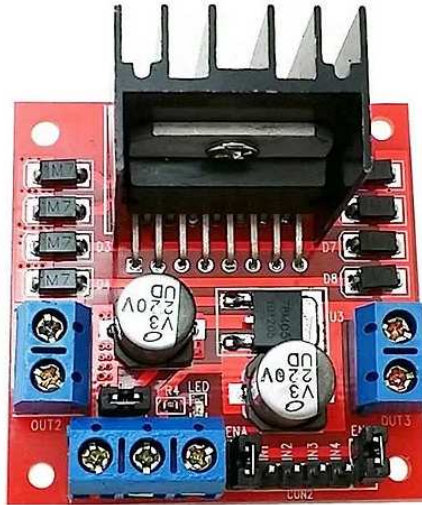
Figura 21 – Garagem simulada utilizada no desafio da aula nove



Fonte: Elaborado pelo autor

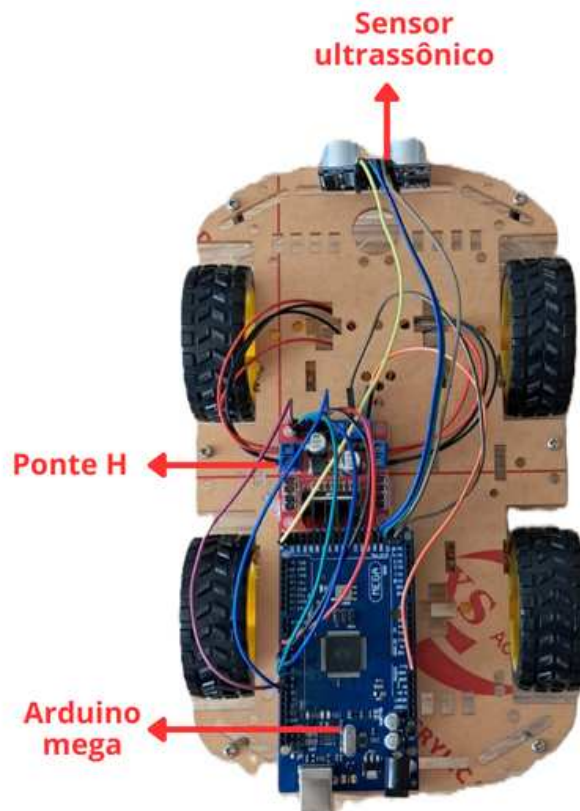
Os robôs ultrassônicos utilizados pelo projeto EPRA são constituídos de um kit chassi 4wd com 4 motores DC, uma ponte H (Figura 22), como controlador um Arduino mega e o sensor, a Figura 23 exhibe a localização dos componentes no robô.

Figura 22 – Ponte H



Fonte: <https://www.roboticaeducacional.art.br/>

Figura 23 – Robô ultrassônico e indicação de componentes



Fonte: Arquivos do projeto EPRA

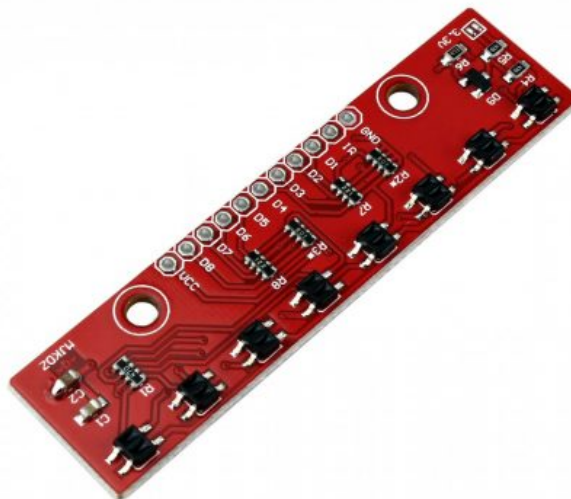
A movimentação do robô acontece através da ponte H, que gerencia os motores. Ela possibilita o controle do sentido em que eles giram e a velocidade, podendo receber alimentação de uma fonte externa como uma bateria.

#### 4.2.4 Parte 4: Projeto Seguidor de Linha

A décima aula marca a parte final do curso, onde foi apresentado o robô seguidor de linha e seus componentes. A aula inicia fazendo a ponte com o módulo anterior, lembrando o trabalho com o robô ultrassônico. Enquanto o último robô reagia a obstáculos pontuais, o seguidor de linha deve realizar uma trajetória definida através da linha, exigindo um controle dos sensores e atuadores mais preciso.

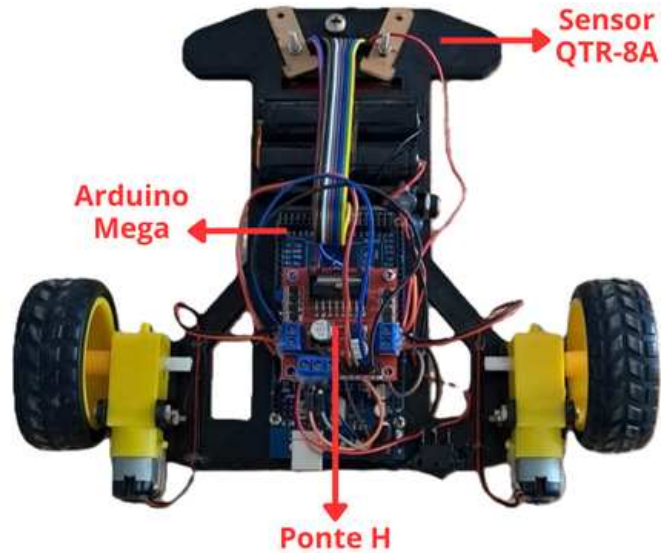
A equipe do projeto EPRA produziu seu próprio modelo de robô seguidor de linha, composto por um chassi impresso em impressora 3d, um arduino mega, motores DC controlados por uma ponte H e os sensores infravermelho. A equipe escolheu utilizar os sensores QTR-8A (Figura 24) que consistem em uma barra com oito sensores IR independentes. A Figura 25 exibe um robô seguidor de linha produzido pela equipe do projeto além de indicar a localização de cada componente.

Figura 24 – Sensor QTR-8A



Fonte: <https://newportcom.com.br/>

Figura 25 – Robô seguidor de linha e indicação de componentes



Fonte: Arquivos do projeto EPRA

A escolha dos sensores infravermelhos QTR-8A foi essencial para o desenvolvimento do desafio, já que através de sua biblioteca dedicada, o controle dos sensores se tornou simples. A partir de um código base (Apêndice B), os alunos apenas deveriam implementar a lógica apresentada em sala de aula.

Na aula 11 os alunos iniciaram a implementação do código para que na aula 12 testassem no circuito. O circuito utilizado no desafio foi providenciado pelo vice coordenador do projeto, exibido na Figura 9.

#### 4.3 Aplicação da proposta e material de apoio

O plano de curso proposto neste trabalho foi aplicado em um curso ofertado pelo projeto EPRA entre novembro e dezembro de 2025, com duração total de 24 horas distribuídas em 12 aulas. Além disso, o autor contou com o apoio de três tutores integrantes da equipe do projeto.

Para a execução das atividades propostas neste plano de ensino, foi desenvolvido um conjunto de recursos didáticos. Foram elaborados slides para guiar as aulas, além da montagem dos circuitos simulados na plataforma Tinkercad, que serviram como base para a compreensão

lógica dos alunos. Também foram desenvolvidos e testados os códigos-fonte utilizados nos robôs ultrassônicos e seguidores de linha, garantindo exemplos funcionais para as práticas.

Com o intuito de facilitar a consulta e permitir a reprodução deste trabalho por outros interessados, todo o material produzido foi organizado em um repositório digital. O acesso completo as apresentações e códigos, encontra-se disponível em **repositório**.

#### **4.4 Método de avaliação**

Para avaliar o aprendizado dos alunos, foram aplicados dois desafios durante o curso. O primeiro foi aplicado na aula 9 onde os alunos deveriam adaptar o código do robô ultrassônico, desenvolvido durante toda parte 3 do curso, fazendo-o entrar e sair da garagem simulada (Figura 21). O segundo desafio foi aplicado na última parte onde os alunos a partir do código base (Apêndice B) deveriam aplicar a lógica apresentada em aula para o robô seguidor de linha e concluir o circuito (Figura 9), iniciando na aula 11 e finalizando na 12.

Além dos desafios, ao final os alunos foram convidados a participar de um questionário baseado na escala Likert, que é muito utilizada em pesquisas das ciências sociais e comportamentais. Ela opera por meio da apresentação de afirmações relacionadas ao assunto de interesse, solicitando aos participantes que manifestem seu grau de concordância em uma série de pontos ordenados, geralmente variando de “discordo totalmente” a “concordo totalmente”. Sua popularidade deve-se principalmente à facilidade de aplicação e interpretação (Júnior, 2014).

Para cada pergunta foi usada a escala de 1 a 7, onde o valor 1 representava o extremo negativo ou de menor intensidade e o valor 7 representava o extremo positivo ou de maior intensidade. Essa métrica permitiu quantificar a percepção dos alunos em relação à eficácia do plano de ensino, o nível de dificuldade enfrentado e a satisfação geral com o aprendizado adquirido. O Apêndice C exibe as perguntas realizadas no questionário.

## **5 ANÁLISE DOS RESULTADOS**

A análise dos resultados será dividido em cinco seções: 1) Análise da utilização da plataforma Tinkercad utilizando dados do formulário; 2) Análise do entendimento dos alunos a respeito da sintaxe da linguagem C, também com o formulário; 3) Análise do entendimento dos alunos a respeito da programação do robô ultrassônico, utilizando o formulário e o desafio 1 (aplicado durante a terceira parte do curso); 4) Análise do entendimento dos alunos a respeito da programação do robô seguidor de linha, utilizando o formulário e o desafio 2 (aplicado durante a quarta e ultima parte do curso); 5) Análise geral.

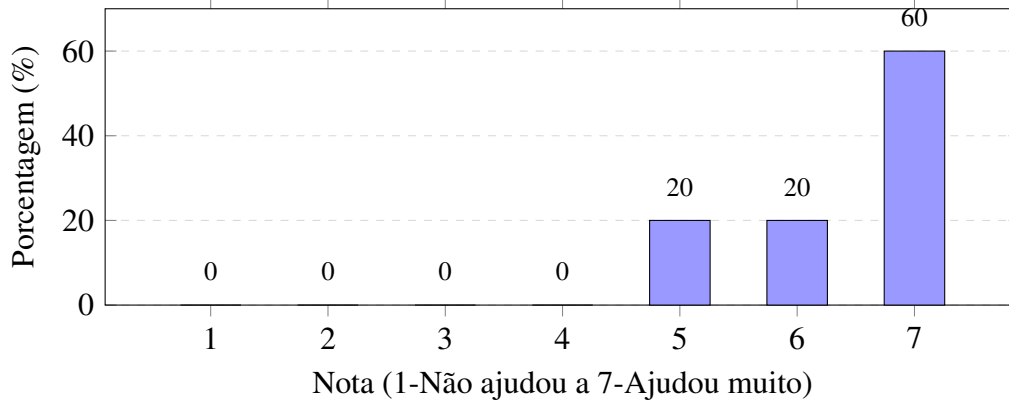
O curso iniciou com 18 alunos inscritos na plataforma tinkercad, finalizando com 10 formados, onde 1 aluno era de ensino médio e 9 discentes da UFC campus Russas, com 6 alunos Engenharia de Software e 3 da Engenharia Mecânica. Todos os 10 alunos responderam o formulário de forma anônima para garantir a total sinceridade deles.

### **5.1 Análise sobre o uso da plataforma Tinkercad**

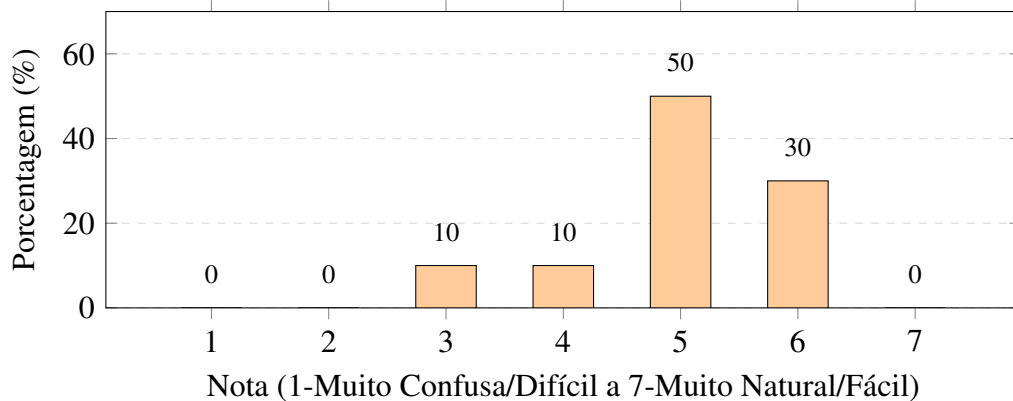
A percepção dos alunos em relação a plataforma Tinkercad, será avaliada considerando as respostas das perguntas 6 e 7 do formulário final. A Figura 26 mostra a distribuição das respostas para cada pergunta.

Figura 26 – Distribuição das respostas para as perguntas relacionadas à plataforma Tinkercad.

(a) O Tinkercad (Simulação) facilitou o entendimento antes de ir para o robô físico (P6)



(b) Dificuldade na transição (P7)



Fonte: Elaborado pelo autor

Na pergunta 6, que questiona o quanto o Tinkercad através das simulações, ajudou os alunos a entenderem os robôs físicos. A maioria dos alunos (80%) atribuiu notas 6 e 7, indicando que a ferramenta foi considerada muito útil, mas há espaço para melhorias, já que 20% dos alunos deram nota 5.

Já na pergunta 7, que avaliou a dificuldade que os alunos sentiram na transição da programação em blocos para a sintaxe em C, 30% dos alunos atribuíram a nota 6, 50% dos alunos nota 5 e 20% notas 3 e 4. Isso sugere que a transição, foi considerada pela maioria dos alunos intermediária/fácil, indicando que há espaço para melhorias devido a tendência para o neutro.

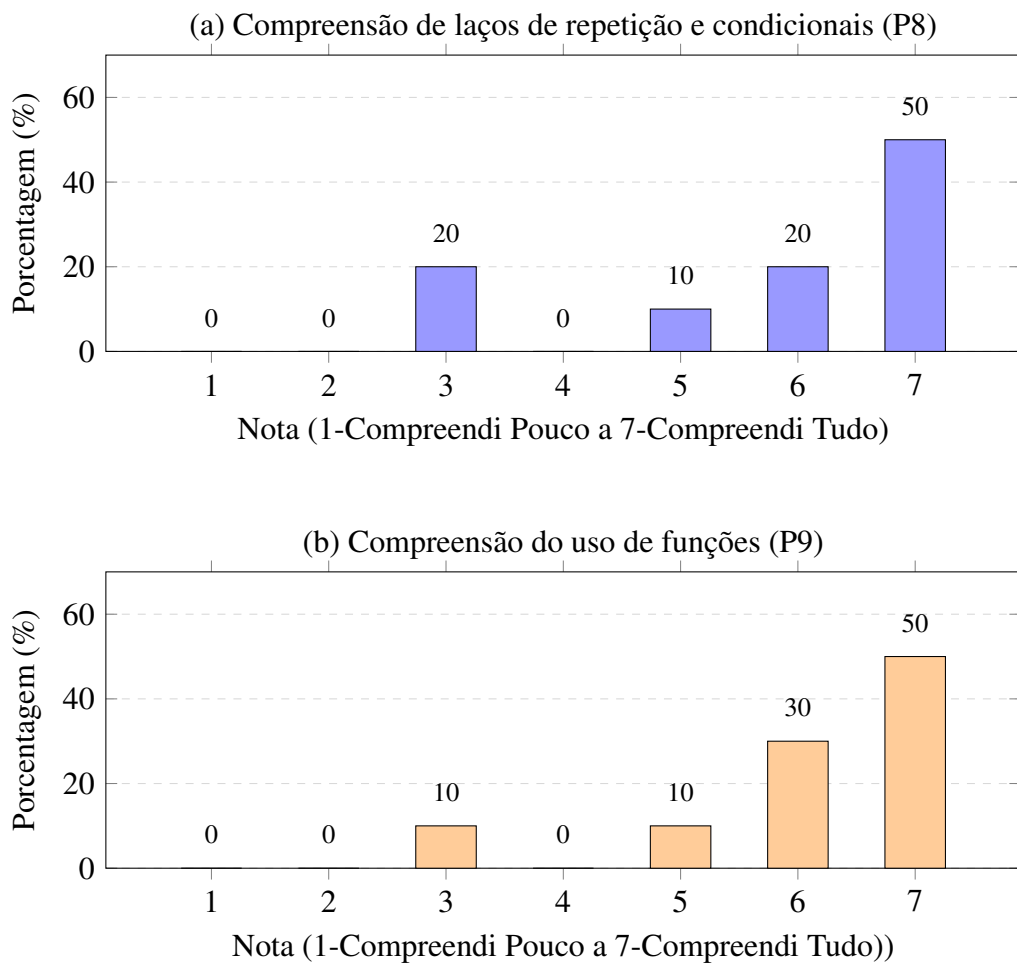
Em resumo, a primeira parte do curso foi bem avaliada, com destaque para a utiliza-

ção do Tinkercad como meio de facilitar o entendimento dos alunos em relação a robótica. Mas que sugere que a transição entre programação em blocos para linguagem de programação ainda pode ser melhorada devido a tendência a neutralidade nas respostas.

## 5.2 Programação em C

A percepção dos alunos em relação ao ensino de programação em linguagem C será avaliada através das perguntas 8 e 9 do formulário final, que abordam a compreensão sobre laços de repetição, condicionais e funções. A Figura 27 mostra a distribuição das respostas para cada pergunta.

Figura 27 – Distribuição das respostas para as perguntas relacionadas à lógica de programação em C.



Fonte: Elaborado pelo autor

Na pergunta 8, sobre a compreensão de laços de repetição e condicionais, a maioria

dos alunos (70%) atribuiu nota 6 e 7 indicando um bom índice de compreensão. A distribuição mostra que 10% dos alunos deram nota 5 e 20% nota 3, indicando que alguns alunos sentiram uma leve dificuldade em compreender o conteúdo das aulas.

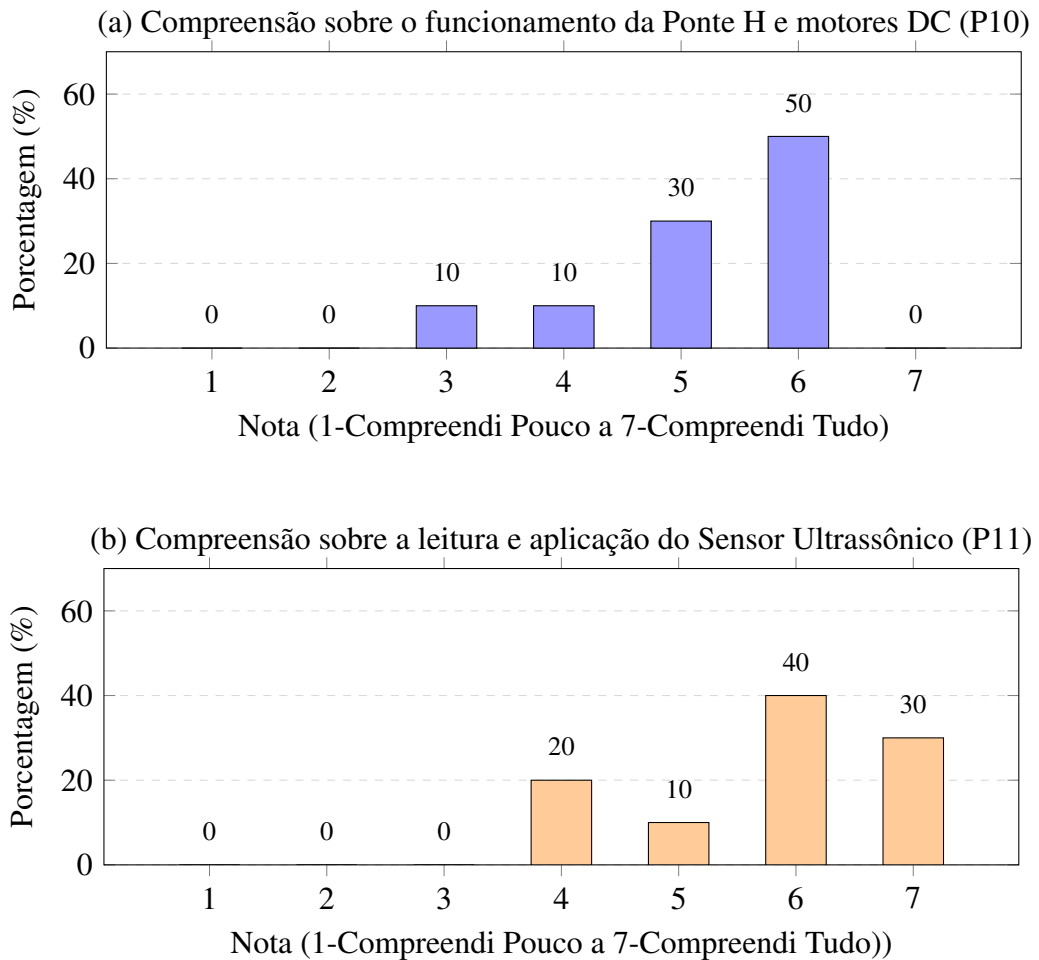
Na pergunta 9, que avaliou a compreensão sobre o uso de funções, a maioria dos alunos (80%) avaliaram com 6 e 7, 10% atribuiu nota 5 e 10% nota 3, indicando também um bom índice de compreensão.

Comparando as duas perguntas, observa-se que a compreensão sobre o uso de funções foi um pouco melhor que a compreensão sobre as estruturas de repetição e condicionais. Em resumo, as aulas que abordaram esses conceitos apresentaram um bom nível de compreensão pelos alunos.

### **5.3 Programação do robô ultrassônico**

A percepção dos alunos em relação a programação do robô ultrassônico será avaliada através das perguntas 10 e 11 do formulário final e o desafio 1. A Figura 28 mostra a distribuição das respostas para cada pergunta do formulário.

Figura 28 – Distribuição das respostas para as perguntas relacionadas aos componentes e lógica do robô ultrassônico.



Fonte: Elaborado pelo autor

Na pergunta 10 que aborda o nível de compreensão dos alunos sobre o uso da ponte H e motores, 50% atribuiu nota 6, 30% nota 5 e 20% para as notas 3 e 4. Indicando que a compreensão foi mediana/boa.

Em relação à compreensão do Sensor Ultrassônico (P11), os resultados foram superiores aos da Ponte H, com 70% dos alunos atribuindo notas 6 e 7. Isso sugere que a lógica de leitura e reação foi assimilada com maior facilidade do que o controle dos motores.

Quanto ao desafio realizado na aula 9, estiveram presentes 7 alunos, onde 6 (divididos em duas equipes de 3) participaram e conseguiram completar o desafio e 1 não demonstrou interesse em participar. Durante a prática, foi notada certa dificuldade nos alunos em como utilizar as estruturas de repetição e condicionais para adaptar o código (desenvolvido por eles ao longo da parte 3) ao desafio, o que entra em concordância com a análise da seção 5.2. Contudo

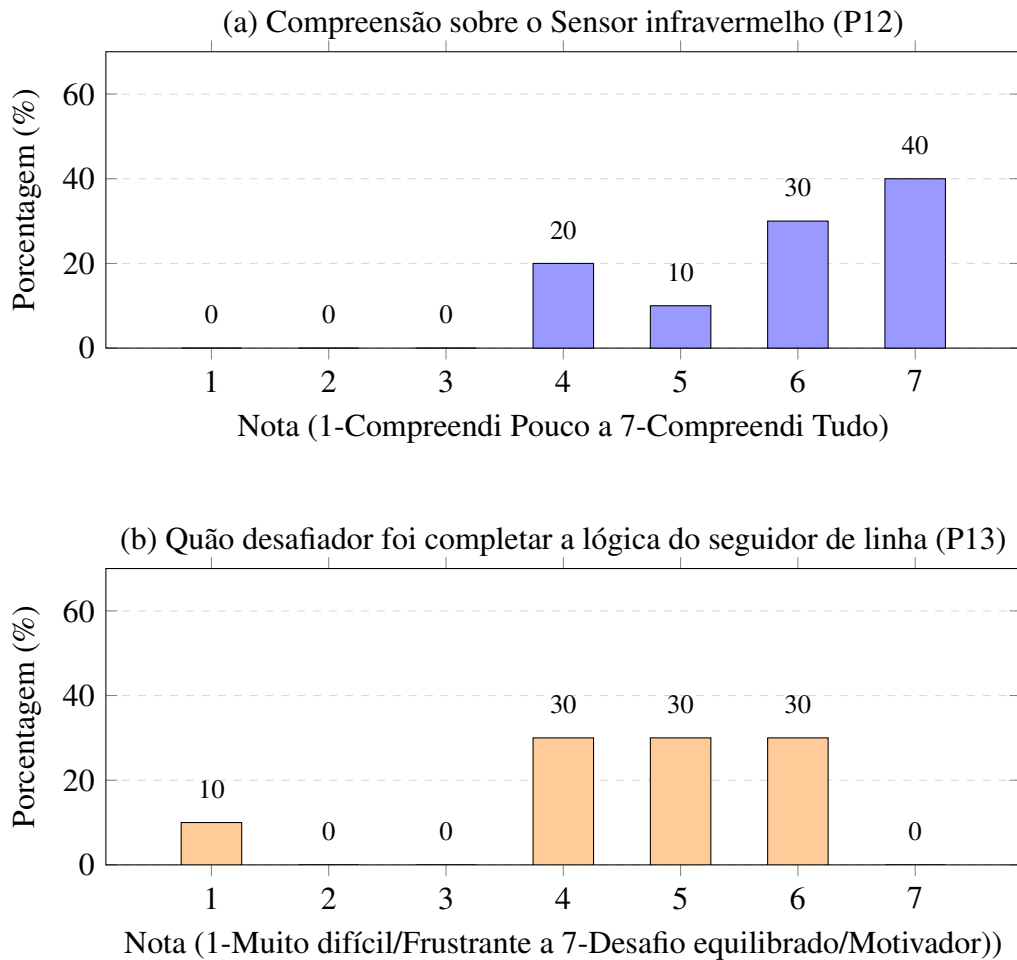
após uma breve revisão sobre essas estruturas, ambas equipes conseguiram finalizar o desafio com sucesso.

E os alunos que não participaram, 1 por desinteresse e 3 por não estarem presentes, só restam suposições sobre o motivo da não participação. Ao considerar o que foi analisado até o momento, o motivo mais plausível seria insegurança sobre a compreensão do que foi estudado, principalmente sobre as estruturas de repetição e condicionais (o que foi apontado até pelas equipes que participaram).

#### **5.4 Programação do robô seguidor de linha**

A percepção dos alunos em relação a programação do robô seguidor de linha será avaliada através das perguntas 12 e 13 do formulário final e o desafio 2. A Figura 29 mostra a distribuição das respostas para cada pergunta do formulário.

Figura 29 – Distribuição das respostas para as perguntas relacionadas aos componentes e lógica do robô seguidor de linha.



Fonte: Elaborado pelo autor

Em relação à compreensão dos alunos sobre o Sensor infravermelho (P12), 70% dos alunos atribuiu notas 6 e 7 (boa compreensão) e 30% notas 4 e 5 (média compreensão). Isso sugere que os alunos compreenderam o sensor infravermelho tão bem quanto o sensor ultrassônico.

Já a pergunta 13 indica complexidade mediana/boa na aplicação da lógica apresentada em sala no código, com 30% dos alunos atribuindo nota 6 e 60% notas 4 e 5. mostrando que mesmo com uma boa compreensão sobre os sensores, ainda existiu certo desafio ao aplicar a lógica no código.

Participaram do desafio 2, ocorrido nas aulas 11 e 12, um total de 8 alunos (divididos em duas equipes de 3 membros e uma de 2). Dois dos alunos que não participaram do ultimo desafio, participaram deste, mostrando que os alunos se sentiram mais motivados no segundo

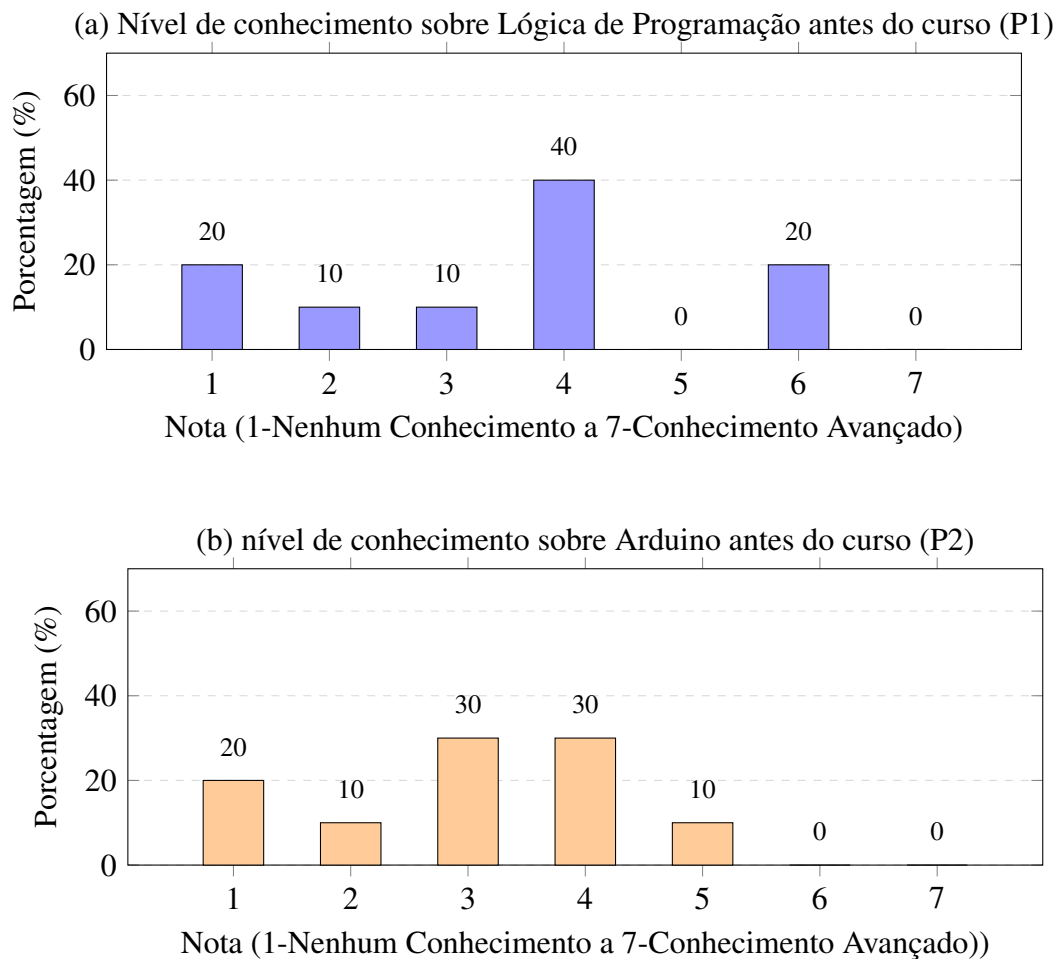
desafio.

Conforme a análise do questionário, os participantes mostraram dificuldade em completar a lógica de controle do robô, sendo necessário a ajuda dos tutores para sanar as dúvidas pontuais que foram surgindo. Contudo, mesmo com dificuldade, todos participantes conseguiram completar o desafio com sucesso.

## 5.5 Análise geral

Para a análise geral, serão usadas as perguntas 1 a 5 do formulário, consideradas perguntas gerais sobre o curso. A Figura 30 exibe a distribuição das respostas para as perguntas relacionadas ao nível de conhecimento dos alunos antes do curso (P1 e P2).

Figura 30 – Distribuição das respostas para as perguntas 1 e 2.



Fonte: Elaborado pelo autor

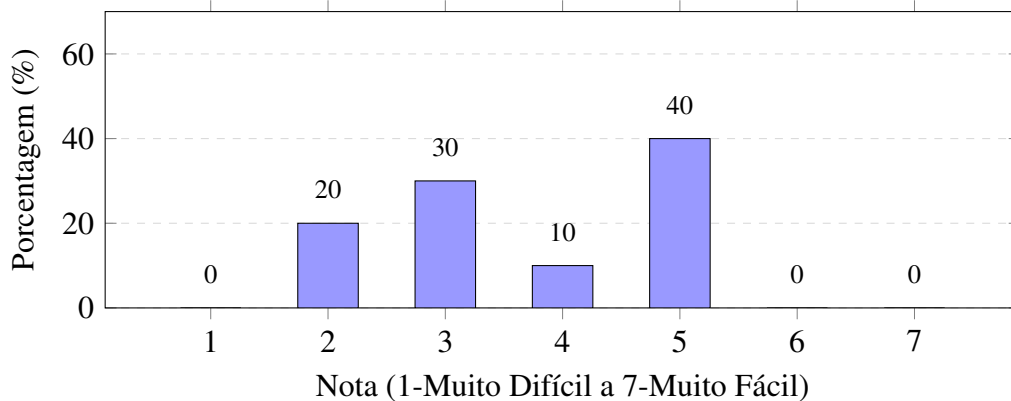
Para a pergunta P1 a respeito do nível de conhecimento sobre Lógica de Programação antes do curso, 40% dos alunos se consideravam de nível intermediário (nota 4), enquanto 30% dizem ter pouco ou quase nenhum conhecimento (nota 1 e 2). Isso mostra que se o curso começasse direto com o ensino da linguagem, os 30% que estão nos níveis 1, 2 (somados) poderiam não acompanhar.

Quanto a P2, ninguém (0%) deu nota 6 ou 7. A maioria (90%) está concentrada nos níveis intermediário/baixo (1, 2, 3 e 4). A nota máxima foi 5 (apenas 10%). Isso mostra que o alunos podiam até saber um pouco de lógica (visto no gráfico anterior), mas não sabiam sobre arduino. Isso valida o uso das simulações com o Tinkercad no início.

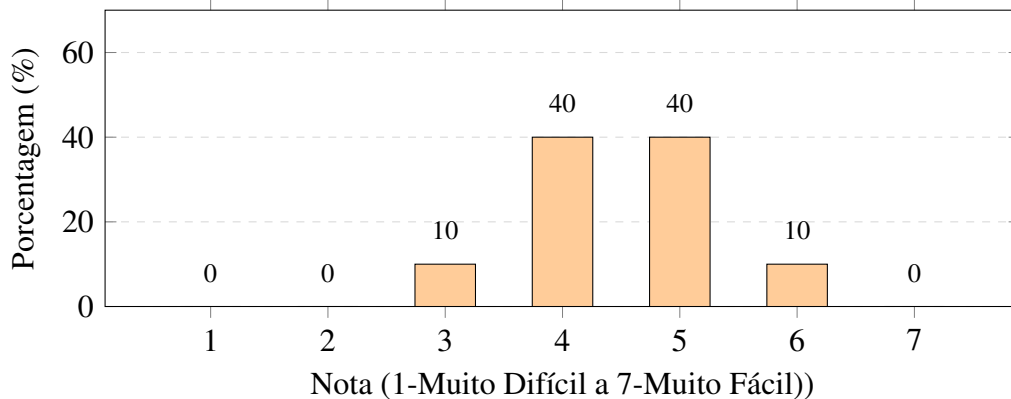
Agora analisando as perguntas (P3 e P4) exibidas na Figura 31, será possível verificar a percepção dos alunos em relação a dificuldade antes e após o curso.

Figura 31 – Distribuição das respostas para as perguntas 3 e 4.

(a) Quão difícil o aluno achava que seria programar em Linguagem C antes do curso (P3)



(b) Após a conclusão, como o aluno avalia a dificuldade real do curso (P4)

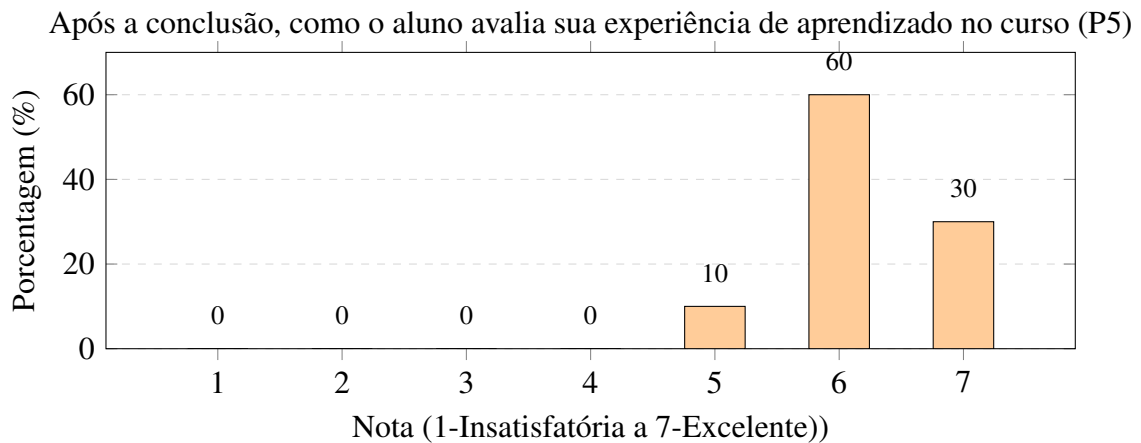


A pergunta P3 colhe as expectativas do aluno quanto a dificuldade esperada antes do curso, onde 40% da turma achava que seria médio/difícil (notas 3 e 4) e 20% que seria difícil. Mostrando que havia um receio quanto a dificuldade que seria encontrada.

Enquanto a P4 colhe a realidade após o curso, com a maioria (80%) concentrando-se nas notas 4 e 5 (médio/fácil). Isso mostra que após a aplicação do plano, a percepção de dificuldade real sofreu uma mudança positiva, com nenhum aluno avaliando como difícil ou muito difícil.

Por último será analisada a pergunta 5 (Figura 32) que busca colher a satisfação geral dos alunos sobre o curso.

Figura 32 – Distribuição das respostas para a pergunta 5.



Fonte: Elaborado pelo autor

Os resultados indicam uma aprovação massiva da metodologia. Não foram registrados índices de insatisfação (notas 1 a 3). A totalidade da turma (100%) avaliou o curso positivamente (notas 5 a 7), com destaque para o fato de que 90% dos alunos atribuíram nota 6 e 7 (“Muito Bom” e “Excelente”).

A correlação entre a redução da dificuldade percebida (Figura 31) e a alta satisfação final (Figura 32) sugere que o plano de ensino foi capaz de tornar o conteúdo técnico acessível sem comprometer a qualidade do aprendizado, engajando os alunos que inicialmente possuíam pouco ou nenhum conhecimento sobre a área.

## 6 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho propôs, preparou, aplicou e validou um plano de ensino estruturado para o ensino de robótica e programação com Arduino, voltado a estudantes do ensino superior. O curso, organizado em quatro partes progressivas do uso do Tinkercad com programação em blocos até a implementação de robôs seguidores de linha demonstrou ser eficaz na introdução de conceitos de lógica, sintaxe C e robótica.

Os resultados, coletados por meio de questionários e desafios, revelaram alta satisfação dos participantes, redução na percepção de dificuldade e um aprendizado significativo, especialmente na etapa final com o robô seguidor de linha. Apesar de desafios como a transição entre blocos e código textual e o controle do robô ultrassônico, a metodologia mostrou-se viável e motivadora.

Como melhoria para o trabalho, a análise dos resultados sugere o refinamento da transição para a linguagem C e do trabalho com o robô ultrassônico. Além disso, surgem como perspectivas futuras, a validação do plano para outras faixas etárias, e como sugestão dos próprios alunos: A exploração de conceitos como Programação Orientada a Objetos (POO) em conjunto com a plataforma Arduino; e o desenvolvimento de projetos relacionados a automação residencial.

Este trabalho reforça assim, o potencial da robótica educacional no âmbito acadêmico como uma ferramenta que estimula a busca por conhecimento e transforma conceitos teóricos em experiências práticas.

## REFERÊNCIAS

- Arduino. **Introdução ao Arduino**. 2018. Disponível em: <https://docs.arduino.cc/learn/starting-guide/getting-started-arduino/>. Acesso em: 14 jul. 2025.
- BRAGA, H. d. C. **A Utilização da Robótica Educacional como Metodologia de Ensino e Aprendizagem**. Trabalho de Conclusão de Curso (Graduação em Engenharia de Controle e Automação) – Universidade Federal de Lavras (UFLA), Lavras, MG, 2020. Disponível em: [https://sip.prg.ufla.br/publico/trabalhos\\_conclusao\\_curso/acessar\\_tcc\\_por\\_curso/engenharia\\_de\\_controle\\_e\\_automacao/index.php?dados=20201201610572](https://sip.prg.ufla.br/publico/trabalhos_conclusao_curso/acessar_tcc_por_curso/engenharia_de_controle_e_automacao/index.php?dados=20201201610572).
- ELETRONICA, S. **Sensor ultrassônico HC-SR04**. 2024. Disponível em: <https://www.sta-eletronica.com.br/artigos/arduinos/sensor-ultrassonico-hc-sr04>. Acesso em: 04 jan. 2026.
- EPPA. **Ensino de Programação e Robótica com Arduino**. 2025. Disponível em: <https://epa.ufc.br/>. Acesso em: 14 jul. 2025.
- JUNIOR, C. R. da S. Robótica nas aulas de matemática do ensino médio: Uma proposta educacional e de baixo custo. **Miscellaneous**, v. 12, n. 5, p. 82–104, 2017.
- JÚNIOR, S. D. d. S. Mensuração e escalas de verificação: uma análise comparativa das escalas de likert e phrase completion. **PMKT-Revista Brasileira de Pesquisas de Marketing, Opinião e Mídia**, ABEP-Associação Brasileira de Empresas de Pesquisa, v. 15, n. 1, p. 1–16, 2014.
- LEITÃO, J. A. Aplicação de robótica computacional no ensino fundamental em crateús. **Trabalho de Conclusão de Curso**, Crateús, CE, Brasil, 2024. Orientador: José Wellington Franco da Silva.
- LIBANEO, J. C. **Didática**. São Paulo, Brasil: Cortez, 2004.
- NASCIMENTO, K. R. S. d. **Formação continuada de professores em Robótica Educacional com práticas no ambiente Tinkercad: uma experiência maker**. Dissertação (Dissertação de Mestrado) – Universidade Federal do Ceará, Programa de Pós-Graduação em Ensino de Ciências e Matemática, Fortaleza, 2024.
- NOGUEIRA, J. A. M. Robótica educacional: Desenvolvimento de atividades práticas utilizando kits de LEGO Mindstorms EV3 para o projeto EPPA. **Trabalho de Conclusão de Curso**, Russas, CE, Brasil, 2024. Orientador: Reuber Regis de Melo.
- PEREIRA, W. G. **Elaboração, construção e execução de um projeto de nivelamento com a finalidade de minimizar a evasão escolar nos cursos de tecnologia**. 2019. 15301–15316 p.
- PUNTES, R. V. **Didática Geral: teorias pedagógicas e planejamento**. Uberlândia, MG: Ed. do Autor, 2023.
- SANT'ANNA, F. M.; ENRICONE, D.; ANDRÉ, L.; TURRA, C. M. **Planejamento de ensino e avaliação**. Porto Alegre: Sagra / DC Luzzatto, 1998.
- SANTIAGO, C. P.; MENEZES, J. W. M.; AQUINO, F. J. A. d. Proposta e avaliação de uma metodologia de aprendizagem baseada em projetos em disciplinas de engenharia de software através de uma sequência didática. **Revista Brasileira de Informática na Educação**, SBC, v. 31, p. 31–59, 2023.

SANTOS, M. D. F. d. S. R. C. **A robótica educacional: entendendo conceitos.** 2020. 345–366 p.

SKOVSMOSE, O. **Educação matemática crítica: a questão da democracia.** Campinas, Brasil: Papyrus, 2001.

SKOVSMOSE, O. **Desafios da reflexão em educação matemática crítica.** Campinas, Brasil: Papyrus, 2008.

SOUZA, P. R. P. D. **Programação e robótica: o impacto da tecnologia na educação / Programming and robotics: the impact of technology on education.** 2021. 82476–82485 p.

VIRMOND, N. A. **PROJETO DE ROBÔ SEGUIDOR DE LINHA.** 2023. 1–13 p.

WILTGEN, F. Robótica prática como ferramenta "mãos à obra" no ensino. **Revista de Engenharia e Tecnologia**, Universidade Estadual de Ponta Grossa, v. 14, n. 2, p. 221–233, 2022. Disponível em: <https://revistas.uepg.br/index.php/ret/article/view/20208>.

## APÊNDICE A – CÓDIGO BÁSICO DE CONTROLE DE UM ROBÔ ULTRASSÔNICO

Código-fonte 1 – Código para controle do robô ultrassônico.

```
1
2 // Definicao dos pinos
3 int velocidadeMotorA = 9;
4 int girarHorarioIN1 = 6;
5 int girarAntiHorarioIN2 = 5;
6 int velocidadeMotorB = 10;
7 int girarHorarioIN3 = 4;
8 int girarAntiHorarioIN4 = 3;
9
10
11 void frente(){
12     analogWrite(velocidadeMotorA,255);
13     analogWrite(velocidadeMotorB,255);
14     digitalWrite(girarHorarioIN1, HIGH);
15     digitalWrite(girarAntiHorarioIN2, LOW);
16     digitalWrite(girarHorarioIN3, HIGH);
17     digitalWrite(girarAntiHorarioIN4, LOW);
18
19 }
20 void virar(){
21     analogWrite(velocidadeMotorA,255);
22     analogWrite(velocidadeMotorB,255);
23     digitalWrite(girarHorarioIN1, LOW);
24     digitalWrite(girarAntiHorarioIN2, HIGH);
25     digitalWrite(girarHorarioIN3, HIGH);
26     digitalWrite(girarAntiHorarioIN4, LOW);
27 }
28
```

```
29 long sensor(){
30     int triggerPin = 52;
31     int echoPin = 51;
32     pinMode(triggerPin, OUTPUT);
33     digitalWrite(triggerPin, LOW);
34     delayMicroseconds(2);
35
36     // Envia um pulso ultrassonico
37     digitalWrite(triggerPin, HIGH);
38     delayMicroseconds(10);
39     digitalWrite(triggerPin, LOW);
40     pinMode(echoPin, INPUT);
41
42     // Mede a duracao do pulso de retorno no pino Echo
43     long duracao = pulseIn(echoPin, HIGH);
44
45     // Calcula a distancia (em cm) baseado na velocidade do
46     som
47     long distancia = duracao * 0.01723;
48
49     return distancia; // Saida em CM
50 }
51
52 void setup()
53 {
54     // Definicao dos pinos como saida
55
56     pinMode(velocidadeMotorA, OUTPUT);
57     pinMode(velocidadeMotorB, OUTPUT);
58     pinMode(girarAntiHorarioIN2, OUTPUT);
59     pinMode(girarHorarioIN1, OUTPUT);
60     pinMode(velocidadeMotor, OUTPUT);
```

```
60     pinMode(girarAntiHorarioIN2, OUTPUT);
61     pinMode(girarHorarioIN1, OUTPUT);
62     Serial.begin(9600);
63
64 }
65
66 void loop(){
67     /* caso a distancia entre o sensor e objeto a frente
68        seja menor que 50cm o robo ira desviar */
69     if(sensor() <= 50){
70         virar();
71     }else{
72         frente();
73     }
74     delay(10);
75 }
```

## APÊNDICE B – CÓDIGO BASE DE CONTROLE DE UM ROBÔ SEGUIDOR DE LINHA

Código-fonte 2 – Código base de controle do robô seguidor de linha.

```
1
2
3
4  #include <QTRsensors.h> /* Biblioteca para leitura dos
   sensores QTR de refletancia (seguidores de linha) */
5
6  /* --- Configuracao dos sensores QTR --- */
7  #define NUM_SENSORS    8          /* Define que serao usados
   8 sensores na barra QTR */
8  #define EMITTER_PIN    2          /* Pino que controla os
   emissores de luz infravermelha dos sensores */
9  #define TIMEOUT        2500      /* Tempo maximo (em
   microssegundos) para leitura do sinal refletido */
10
11
12
13  QTRsensors qtr;                  // Cria um objeto da
   biblioteca QTR para manipular os sensores
14  uint16_t sensorValues[NUM_SENSORS]; // Vetor que
   armazenara as leituras de cada sensor
15
16  //
17  int girarHorarioIN1 = 5;
18  int girarAntiHorarioIN2 = 6;
19  int girarHorarioIN3 = 9;
20  int girarAntiHorarioIN4 = 10;
21
22
```

```
23 // --- Funcao para controle dos motores com ponte H
24 void frente(int VEL_A,int VEL_B){
25
26     analogWrite(girarHorarioIN1, VEL_A);
27     analogWrite(girarAntiHorarioIN2, 0);
28     analogWrite(girarHorarioIN3, VEL_B);
29     analogWrite(girarAntiHorarioIN4, 0);
30
31 }
32
33
34 int velocidadeBase = 120;      // Velocidade media usada
    em linha reta
35 float ganho = 0.05;           // Fator de correcao
    proporcional (sensibilidade de curva)
36
37
38
39
40 void setup() {
41
42     // --- Inicializa e configura os sensores QTR ---
43     qtr.setTypeRC();
44     qtr.setSensorPins((const uint8_t[]){34, 35, 36, 37, 38,
45         39, 40, 41}, NUM_SENSORS);
46
47     qtr.setEmitterPin(EMITTER_PIN);
48
49     // Calibracao dos sensores
50     pinMode(LED_BUILTIN, OUTPUT);
51     digitalWrite(LED_BUILTIN, HIGH);
52     for (uint16_t i = 0; i < 400; i++) qtr.calibrate();
53     digitalWrite(LED_BUILTIN, LOW);
```

```
52
53 // Definicao dos pinos como saida
54 pinMode(girarAntiHorarioIN2, OUTPUT);
55 pinMode(girarHorarioIN1, OUTPUT);
56 pinMode(girarAntiHorarioIN4, OUTPUT);
57 pinMode(girarHorarioIN3, OUTPUT);
58
59
60 }
61
62 void loop() {
63
64     uint16_t posicao = qtr.readLineBlack(sensorValues);
65     // Retorna valor de 0 a 7000 (cada sensor representa
66         1000 unidades)
67
68     delay(10); // Pequeno delay para estabilidade
69
70 }
```

## **APÊNDICE C – QUESTIONÁRIO PARA AVALIAÇÃO DO PLANO DIRECIONADO AOS ALUNOS**

**Questão 1.** Qual era seu nível de conhecimento sobre Lógica de Programação antes deste curso?

**Questão 2.** Qual era seu nível de conhecimento sobre Arduino antes deste curso?

**Questão 3.** Quão difícil você achava que seria programar em Linguagem C antes de começar?

**Questão 4.** Após a conclusão, como você avalia a dificuldade real do curso?

**Questão 5.** De modo geral, como você avalia sua experiência de aprendizado no curso?

**Questão 6.** O uso inicial do Tinkercad (Simulação) facilitou o entendimento antes de ir para o robô físico?

**Questão 7.** Como você avalia a dificuldade na transição da programação em Blocos para a sintaxe em C

**Questão 8.** Quanto você compreendeu sobre o funcionamento de Laços de Repetição (for, while) e Condicionais (if, else) na linguagem C?

**Questão 9.** Quanto você compreendeu sobre o uso de Funções para organizar o código?

**Questão 10.** Quanto você compreendeu sobre o funcionamento da Ponte H para controlar os Motores DC?

**Questão 11.** Quanto você compreendeu sobre a leitura e aplicação do Sensor Ultrassônico?

**Questão 12.** Quanto você compreendeu sobre a leitura e aplicação do Sensor infravermelho (seguidor de linha)?

**Questão 13.** Quão desafiador foi completar a lógica do seguidor de linha a partir do código base?