



OPERAÇÕES MATRICIAIS MEDIANTE A MOVIMENTAÇÃO DE UM VEÍCULO GUIADO AUTOMATICAMENTE

GEOVANA FERREIRA DO NASCIMENTO

Trabalho de Conclusão do Curso Superior de Licenciatura em Matemática, orientado
pelo Prof. Dr. Marco Aurélio Granero Santos.

IFSP

São Paulo

2018

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO
PAULO

Câmpus São Paulo

GEOVANA FERREIRA DO NASCIMENTO

**OPERAÇÕES MATRICIAIS MEDIANTE A MOVIMENTAÇÃO DE UM VEÍCULO
GUIADO AUTOMATICAMENTE**

Trabalho de Conclusão de Curso apresentado ao Instituto de Educação, Ciência e Tecnologia de São Paulo, orientado pelo Prof. Dr. Marco Aurélio Granero Santos, em cumprimento ao requisito parcial para obtenção do grau acadêmico de Licenciada em Matemática.

São Paulo

2018

Catalogação na fonte
Biblioteca Francisco Montojos - IFSP Campus São Paulo
Dados fornecidos pelo(a) autor(a)

N244o	<p>Nascimento, Geovana Ferreira do Operações matriciais mediante a movimentação de um veículo guiado automaticamente / Geovana Ferreira do Nascimento. São Paulo: [s.n.], 2018. 95 f.</p> <p style="text-align: center;">Orientador: Marco Aurélio Granero Santos</p> <p style="text-align: center;">Trabalho de Conclusão de Curso (Licenciatura em Matemática) - Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, IFSP, 2018.</p> <p style="text-align: center;">1. Transformações Lineares. 2. Veículo Guiado Automaticamente. 3. Arduino. I. Instituto Federal de Educação, Ciência e Tecnologia de São Paulo II. Título.</p> <p>CDD 510</p>
-------	---

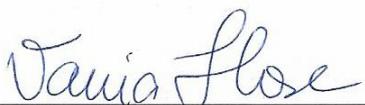
GEOVANA FERREIRA DO NASCIMENTO

**OPERAÇÕES MATRICIAIS MEDIANTE A MOVIMENTAÇÃO DE
UM VEÍCULO GUIADO AUTOMATICAMENTE**

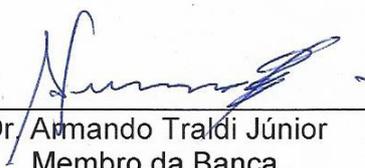
Monografia apresentada ao Instituto Federal de
Educação, Ciência e Tecnologia de São Paulo, em
cumprimento ao requisito exigido para a obtenção do
grau acadêmico de Licenciado em Matemática.

APROVADO EM 10/12/2018

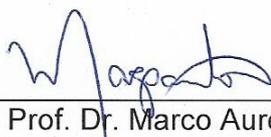
CONCEITO: Doz



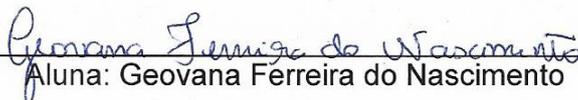
Me. Vania Batista Flose Jardim
Membro da Banca



Dr. Armando Traldi Júnior
Membro da Banca



Prof. Dr. Marco Aurélio Granero Santos
Orientador



Aluna: Geovana Ferreira do Nascimento

“No mundo das tecnologias, o papel do professor será mais valorizado, como formador na ética e na cidadania, o que nenhuma máquina pode fazer”.

Andrea Ramal

Los Meus Pais

AGRADECIMENTOS

Agradeço primeiramente a Deus, pela força e coragem ao longo desta trajetória, aos professores que me acompanharam durante a graduação, em especial ao Professor Dr. Marco Aurélio Granero Santos pela orientação, paciência e confiança que tornaram possível a conclusão deste trabalho. Agradeço também a meu pai, Antonio, que não mediu esforços para me ajudar a alcançar meus objetivos e sempre foi uma fonte de incentivo e inspiração. À minha mãe, Marilda e a meu irmão, Gustavo, pelo apoio e incentivo nos momentos difíceis. E por último, mas não menos importante, aos amigos que fiz na Licenciatura pela companhia e encorajamento de sempre.

RESUMO

Este trabalho relaciona a Robótica, a Álgebra Linear e a Linguagem de programação envolvidas na movimentação de um Veículo Guiado Automaticamente no plano cartesiano. Assim, o trajeto do veículo é ferramenta para o estudo de transformações geométricas por meio das operações envolvendo matrizes, e a lógica de programação é abordada por meio da programação do Arduino que permite a movimentação do Veículo.

Palavras-chaves: Arduino; Veículo Guiado Automaticamente; Transformações Lineares.

ABSTRACT

This work relates Robotics, Linear Algebra and the Programming Language involved in the movement of an Automatic Guided Vehicle on the Cartesian plane. Thus, the vehicle's path is a tool for the study of geometric transformations by means of operations involving arrays, and the programming logic is addressed through the Arduino programming that enables movement of the vehicle.

Keywords: Arduino; Automatic Guided Vehicle; Linear Algebra.

LISTA DE FIGURAS

Figura 1: Diferentes tipos de Veículos Guiados Automaticamente.....	28
Figura 2: Chassi semelhante ao fornecido pelo fabricante.....	29
Figura 3: Suporte para baterias.....	30
Figura 4: Pinos de alimentação e conector de entrada do Arduino.....	31
Figura 5: Conexão USB do Arduino.....	31
Figura 6: Conector de alimentação DC.....	31
Figura 7: Rodas e motores com caixa de redução e eixo duplo.....	32
Figura 8: Lista de parafusos e porcas utilizadas.....	33
Figura 9: Sensor fornecido pelo fabricante.....	34
Figura 10: Módulo fornecido pelo fabricante.....	34
Figura 11: Arduino UNO R3.....	35
Figura 12: Layout da placa e pinos do Arduino Uno R3.....	37
Figura 13: Instalação dos motores, driver do motor e conexões.....	38
Figura 14: Instalação do Arduino e sua fonte de alimentação,.....	39
Figura 15: Conexões entre o suporte da bateria, o Arduino e o driver do motor.....	39
Figura 16: Conexões entre o driver do motor e o medidor de voltagem.....	40
Figura 17: Conexões entre o Arduino e o driver do motor.....	40
Figura 18: Fotografia do veículo após a finalização da primeira lição.....	41
Figura 19: Instalação dos sensores.....	41
Figura 20: Instalação dos sensores de rastreamento e conexões entre o Arduino e os sensores.....	42
Figura 21: Diagrama de Venn.....	44
Figura 22: Outros exemplos de funções ou transformações. Fonte: Diagrama de Venn elaborada com a ferramenta para inserir formas no Microsoft Word.....	45
Figura 23: Exemplos de relação que não são funções ou transformações. Fonte: Diagrama de Venn elaborada com a ferramenta para inserir formas no Microsoft Word.....	45
<i>Figura 24: Quadrado formado pelos vetores u, v, t e w e quadrado formado pelos vetores transformados. Fonte: Elaborada no Geogebra.....</i>	<i>47</i>
Figura 25: Quadrado formado pelos vetores A e B.....	49
<i>Figura 26: Figura elaborada no Geogebra para exemplificar as transformações $T_1, T_2,$ e T_3.....</i>	<i>50</i>

Figura 27: Quadrado formado pelos vetores A e B.	52
Figura 28: Figura elaborada no Geogebra para exemplificar as transformações 4, 5 e 6.	52
Figura 29: Quadrado que será rotacionado.	53
Figura 30: Imagem da transformação dos vetores cujas extremidades são A,B, C e D. Fonte: Elaborado no Geogebra.....	54
Figura 31: Figura elaborada no Geogebra para exemplificar as Transformações de 7 a 13.	57
Figura 32: Figura da transformação de rotação de uma figura no centro do sistema de coordenadas.....	57
Figura 33: Figura elaborada no Geogebra para exemplificar reflexões ao redor dos eixos.	59
Figura 34: Figura Elaborada no Geogebra para exemplificar a reflexão em torno da origem.....	60
Figura 35: Figura elaborada no Geogebra como suporte para exemplificar uma reflexão.....	61
Figura 36: Figura para exemplificar os cisalhamentos.	62
Figura 37: Fluxograma que descreve o funcionamento do IF.....	65
Figura 38: Fluxograma que mostra a ordem e o que a programação executa.	67
Figura 39: Fluxograma que mostra o que a função Rastreamento.....	68
Figura 40: Representante do Vetor AB com extremidade na origem do plano cartesiano.....	69
Figura 41: Movimentação por uma faixa na horizontal.	71
Figura 42: Movimentação do veículo sob uma faixa vertical.	72
Figura 43: Movimentação do veículo na circunferência.....	73
Figura 44: Parte da movimentação descrita por transformações compostas.	78

LISTA DE TABELAS

Tabela 1: Conexões entre o Arduino e os sensores.....	42
Tabela 2: Tipo de dados e valores que podem ser atribuídos a eles.	64
Tabela 3: Valor obtido pela leitura do pino e seu significado.....	64
Tabela 4: Configuração dos pinos com a função "pinMode".	65
Tabela 5: Operadores lógicos na linguagem C/C++ e seus significados.....	66

SUMÁRIO

i1. INTRODUÇÃO	21
1.2. METODOLOGIA	25
2. VEÍCULO GUIADO AUTOMATICAMENTE	27
2.1. Carro Robótico OSOYOO	28
2.1.1. Principais componentes do kit	29
2.1.2. Montagem	37
3. ÁLGEBRA LINEAR	43
3.1 Espaço vetorial	43
3.2 TRANSFORMAÇÕES	44
3.2.2 Transformação Linear	46
3.3 TRANSFORMAÇÕES GEOMÉTRICAS NO PLANO	46
3.2.1 Dilatação	47
3.3.2. Contração	50
3.2.2 Rotação	52
3.2.3 Reflexão	58
3.2.4 Cisalhamento	61
4. PROGRAMAÇÃO	63
4.1. Principais comandos	63
4.1.1. Desvio condicional	65
4.2. Fluxograma	66
5. MATRIZES NA DESCRIÇÃO DO TRAJETO PERCORRIDO POR UM AGV	69
5.1. Dilatação	69
5.2. Rotação	72
5.3. Reflexão	73
5.4. Transformações compostas	75
5.4. Programação e algoritmo	78
Anexo A - Programa comentado do Veículo Guiado Automaticamente	83
REFERÊNCIAS	91

1. INTRODUÇÃO

Vivemos uma época sem precedentes. Schwab (2016, apud TAMASHIRO; GANOKA; CARDOSO, 2017) apontou que a humanidade nunca passou por um momento “tão promissor ou perigoso” quanto o presente. O aumento da utilização e da aceitação da inteligência artificial, o desenvolvimento e a aplicação da robótica em diversas áreas, a evolução e a acessibilidade da internet além da expansão e do avanço da nanotecnologia evidenciam que vivemos no que Schwab (2016) chama de Quarta Revolução Industrial. Essa Revolução afeta nossas vidas em todos os aspectos, entre eles a educação.

Tendo em vista o momento de ascensão tecnológica que presenciamos, é imprescindível que o professor utilize recursos tecnológicos e didáticos compatíveis ao período histórico que vivemos.

Uma questão que gera frequente preocupação entre os professores é a utilização de métodos não tradicionais para ensinar os conteúdos do currículo de forma atrativa e cativante. Uma vez que, além de atender o conhecimento específico da área ministrada o educador também deve assegurar uma formação completa e ao estudante. Segundo Panciera e Ferreira (2006), diferentes aplicabilidades devem ser apresentadas aos alunos para “contribuir para sua formação integral para a vida e para o trabalho”, e ainda sobre o papel significativo que a educação deve ter na vida do aluno Sanches (2002) afirma:

O movimento de educação matemática em todo o mundo levou a uma série de questionamentos sobre o papel formativo e instrumental da Matemática; passou-se a discutir a necessidade de que o ensino da Matemática privilegiasse a formação global do indivíduo com vistas a uma maior inserção social. (Imenes, 1989, D'Ambrosio, 1990; apud SANCHES, 2002, p.12)

De acordo com Fagundes (2005), para transformar a vida do aluno, o educador deve desafiá-lo, propondo problemas que exijam soluções criativas e inovadoras. Para ensinar de forma interessante os conteúdos, o professor deve utilizar diferentes meios e recursos a fim de proporcionar uma formação atual, efetiva e compatível com realidade em termos de tecnologia, por isso, muitos professores têm recorrido aos recursos tecnológicos.

A robótica educacional tem sido apontada como uma excelente alternativa para o ensino de conteúdos da Matemática e da Física de forma moderna e prazerosa. Chella (2002) designa a robótica educacional como sendo

[...] o ambiente constituído pelo computador, componentes eletrônicos, eletromecânicos e programa, onde o aprendiz, por meio da integração destes elementos, constrói e programa dispositivos automatizados com o objetivo de explorar conceitos das diversas áreas do conhecimento. (CHELLA, 2002, p. 23)

De acordo com Chella (2002) e Torcato (2012), a robótica educacional possibilita o estudo de assuntos multidisciplinares da Matemática, Física e Informática fazendo uso de conceitos, até então, restritos à robótica industrial. De fato, a robótica educacional contribui para a integração entre o que há de concreto e abstrato nos conteúdos, ajuda o aluno a desenvolver o raciocínio lógico e a se esforçar na busca de soluções para situações do cotidiano, promovendo a autonomia e viabilizando a aprendizagem prática. Ela também propicia a inclusão tecnológica, permitindo que o educando se mantenha atualizado por conhecer recursos tecnológicos modernos e consiga interagir melhor com o mundo na sua formação tecnológica contemporânea.

Por ser desafiadora e lúdica, a robótica educacional, de acordo com Oliveira (2013), oportuniza uma aprendizagem relevante dos conceitos e relações matemáticas envolvidas no objeto de estudo, o que torna cada aluno participante ativo de seu próprio processo de aprendizagem. Outrossim, a aplicação da teoria matemática na robótica, não apenas propicia a aprendizagem dos conteúdos, mas também promove concepções que o currículo não abrange, como o trabalho em equipe, autodesenvolvimento, senso crítico, integração das disciplinas, exposição e defesa de pensamentos, criatividade, autonomia e responsabilidade, conforme argumentam Trentin, Pérez e Teixeira (2013).

Infelizmente, boa parte da tecnologia usada na robótica educacional ainda possui um custo elevado, sendo assim, de difícil acesso, levando-se em consideração a realidade das escolas brasileiras. Por isso, surgiu o conceito de Robótica Educacional Livre, que de acordo com Trentin, Pérez e Teixeira (2013), visa dar às escolas a oportunidade de ter acesso à robótica de baixo custo, que não necessita de um grande investimento inicial porque utiliza Softwares e Hardwares Livres.

A Robótica Educacional Livre faz com que o ensino se torne mais acessível à maioria das escolas o que contribui para o desenvolvimento de diversas competências e habilidades importantes para a formação do educando. Wing, *apud* Oliveira (2014), argumenta a fim de destacar a importância e os benefícios da programação no ensino dizendo:

[...] esse subconjunto de competências e habilidades pode ser denominado como pensamento computacional, envolvendo as competências relacionadas à abstração e decomposição de problemas de forma a permitir sua resolução usando recursos computacionais e estratégias algorítmicas. (OLIVEIRA *et al.*, 2014, p.584)

Logo, a utilização de ferramentas algorítmicas incorporadas na programação, permite ao aluno aprender com a experimentação, com a resolução de problemas pela análise de erros e acertos e com a aplicação do conceito estudado em situações concretas. Além disso, ainda permite ao educando, aprimorar suas habilidades e seus conhecimentos acerca dos fundamentos da computação, conhecimentos esses que devem ser desenvolvidos ainda na educação básica segundo a Sociedade Brasileira de Computação (SBC, *apud* OLIVEIRA *et al.*, 2014).

Com base nos conceitos descritos anteriormente, este trabalho proporciona uma abordagem exploratória das Matrizes por meio da robótica. Posto que Sanches (2002) afirma que nas aulas de matemática, as matrizes

[...] aparecem apenas como ferramentas ágeis na discussão e resolução de sistemas lineares, não sendo apresentada nenhuma sugestão quanto à introdução do conceito através de situações-problema e também não é feita qualquer menção à história do surgimento do conceito de matrizes. (SANCHES, 2002, p.6)

Considerando a análise de Sanches quanto à maneira como tem sido conduzido o ensino de matrizes, percebe-se que o aprendizado desse conteúdo na Educação Básica normalmente é aplicado apenas ao cálculo de determinantes e resolução de sistemas lineares, de forma desconectada e sem aplicação prática em outras áreas. Por essa razão, este trabalho visa aplicar os conhecimentos associados às matrizes à movimentação de um veículo no plano. Uma vez que o ensino tradicional deste assunto no Ensino Médio não visa à contextualização com momento histórico atual mediante a quarta revolução industrial ou a aplicação do conteúdo no cotidiano do educando.

Muitos pesquisadores descrevem, sem finalidade pedagógica, a movimentação de objetos no plano cartesiano e no espaço usando ferramentas matemáticas. Assim como relata Mesquita (2016), a maioria deles traçam esquemas sobre braços robóticos e os referenciais necessários para calcular a posição final do seu punho usando matrizes. Entretanto, quando consideramos as várias linhas de pesquisas existentes para propor a utilização da robótica educacional no ensino de matrizes, constata-se que a maioria delas utiliza um braço mecânico para estudar sua movimentação no espaço. Uma dessas pesquisas foi proposta por Silva (2017) com a dissertação intitulada: "Utilizando o Arduino como atividade aberta de investigação e experimentação matemática para o ensino de conceitos de matrizes" que apresenta possibilidades para o ensino de matrizes utilizando o Arduino no controle de um painel de LEDs e um braço robótico.

Assim como Silva (2017), diversos pesquisadores buscam metodologias para impulsionar o aprendizado dos alunos ao estudarem conteúdos sobre Programação e Física usando o Arduino. Macedo e Prietch (2013), por exemplo, usam o Arduino como ferramenta de apoio ao ensino de programação por propor e analisar os resultados acerca do curso de extensão chamado "Aprendendo a programar com LEDs". Enquanto Souza *et al* (2011), apresenta o Arduino como "uma opção de baixo custo para experiências de Física assistidas pelo PC" e se encaixa no seguimento de pesquisas que propõem experiências didáticas de Física assistidas por microcontroladores.

Contamos também com estudos como o de Silva e Scherer (2013) que apontam a robótica como uma excelente ferramenta para "trabalhar conceitos ligados a matérias curriculares tais como Física e Matemática". Uma delas foi proposta por Delazeri *et al* (2015) e trata do "desenvolvimento de um carrinho autônomo com Arduino para o ensino de Cinemática", trazendo como solução didática e pedagógica para ensinar as Leis da Física, Movimento Uniforme e Movimento Uniformemente Variado a utilização de um veículo com as mesmas características do veículo usado neste trabalho.

A Matemática e a Física ainda podem ser trabalhadas juntas utilizando a robótica como instrumento. Por esta razão também existem diversas propostas para a abordagem interdisciplinar da Matemática com a Física, que tratam de conteúdos

como funções e movimento retilíneo uniforme utilizando um Veículo Guiado Automaticamente. Além disso, umas das atividades propostas por Pantoja (2016) aplica as operações matriciais no trajeto de um veículo robótico também controlado via programação em Arduino, no entanto o veículo não é um Guiado Automaticamente, ele se desloca por um percurso previamente definido.

Doravante, apesar de todo o estudo já realizado envolvendo a álgebra linear e a movimentação via microcontroladores, o estudo do trajeto percorrido por um veículo com essas características no plano cartesiano por meio das operações matriciais ainda é algo pouco explorado.

1.2. METODOLOGIA

Considerando que este trabalho visa explorar os conceitos de operações matriciais por meio da movimentação de um Veículo Guiado Automaticamente e tendo em vista a definição de autores como Gil (2002), Godoy (1995), Neves (1996), e Vilaça (2012) quanto a alguns aspectos da metodologia de pesquisa, este trabalho possui finalidade exploratória, uma vez que pretende explorar um conceito matemático, as matrizes, mediante a movimentação do veículo.

De acordo com Gil (2002) o objetivo de uma pesquisa exploratória é:

[...] proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a constituir hipóteses. Pode-se dizer que estas pesquisas têm como objetivo principal o aprimoramento de ideias ou a descoberta de intuições. (GIL, 2008, p.41)

Esta pesquisa é ainda de natureza aplicada porque segundo Seliger & Shohamy (2001), Gil (2002) e Richardson (2007) citados por Vilaça (2012), ela observa e descreve comportamentos explorando um aspecto pouco conhecido, uma vez que descreve a movimentação observada por meio de operações com matrizes.

Para apresentar os conceitos e a metodologia de pesquisa utilizada para atingir os objetivos propostos, este trabalho será dividido em cinco capítulos, incluindo esta introdução e das considerações finais. No capítulo 2 são descritos os materiais utilizados e o funcionamento dos principais componentes. No capítulo 3 a teoria Matemática envolvida. No capítulo 4 são apresentas a programação. E por fim

no capítulo 5 é esquematizada a relação entre a linguagem de programação, a álgebra linear e a movimentação do veículo.

2. VEÍCULO GUIADO AUTOMATICAMENTE

Os Veículos Guiados Automaticamente, ou simplesmente AGV's, do inglês *Automated Guided Vehicle*, são máquinas que tem conquistado cada vez mais espaço nas indústrias da atualidade. De acordo com Silva (2010), Mestre em Engenharia de Automação e Sistemas, os AGVs são

[...] veículos que se deslocam de modo autônomo, gerenciados por um sistema de controle, através de um mapa de percurso pré-definido. (SILVA, 2010, p. 23)

Em outras palavras, a principal característica de equipamentos como estes é a capacidade de receber e executar instruções sem a existência de um condutor. Tais veículos conseguem transportar materiais dentro de uma indústria sendo guiados apenas por mecanismos indutivos: obtidos por meio da instalação de fios condutores no piso; por navegação ótica: mediante uma faixa que pode ser construída com pintura ou fita adesiva, ou ainda, via tecnologia a laser: com ajuda de espelhos reflexivos.

Veículos com essas particularidades se tornaram populares nas indústrias por apresentarem vantagens convenientes ao processo de produção. Entre essas vantagens estão: a diminuição dos custos com manutenção, aumento da produtividade e flexibilidade ao processo, uma vez que têm a capacidade de se adaptar facilmente as mudanças no trajeto e são capazes de manusear diversos tipos de cargas.

Estes veículos podem ser classificados de acordo com uso e a carga que transportam, como mostra a Figura 1. São chamados de veículos rebocadores, quando são carregados e descarregados nos postos de coleta e percorrem um trajeto predefinido dentro da fábrica; veículos para carga unitária, se carregam caixas em plataformas de madeira, metal ou plástico; veículos empilhadeira caso, sem a ajuda de um operador, realizem a mesma função da empilhadeira tradicional ou ainda podem ser caracterizados como carrinhos automaticamente guiados se forem projetados para o transporte de cargas menores.



Figura 1: Diferentes tipos de Veículos Guiados Automaticamente.
Fonte: <https://www.imam.com.br/consultoria/artigo/pdf/fundamentos-veiculos-automaticamente-guiados.pdf>

Apesar de *Veículo Guiado Automaticamente* ser um termo que descreve máquinas industriais que apresentam as características e as vantagens citadas acima, neste trabalho usamos um veículo com as mesmas propriedades de um AGV, uma vez que tratamos da movimentação de um veículo robótico controlado por navegação ótica via programação em Arduino, mas diferentemente das aplicações industriais, vamos usá-lo para explorar conteúdos matemáticos e realizar operações matriciais baseadas na movimentação deste veículo.

2.1. Carro Robótico OSOYOO

Existem atualmente no mercado, diversos modelos de veículos robóticos que atenderiam as necessidades deste trabalho. No entanto, para a realização dos testes, escolhemos o *Robot Car Starter Kit* fabricado pela OSOYOO. Esse kit é

apropriado por diversos motivos. O veículo é controlado por Arduino, um hardware livre. Ele possui cinco sensores seguidores de linha, o que permite que ele se mova a um ângulo maior do que veículos deste tipo habitualmente se movem, já que na maioria das vezes tais veículos possuem apenas três módulos. O kit ainda é fácil de manusear e autoexplicativo, pois foi projetado para ser usado por iniciantes na programação Arduino, por isso possui tutoriais detalhados sobre a montagem, além de disponibilizar os gráficos de circuito e os códigos de programação.

2.1.1. Principais componentes do kit

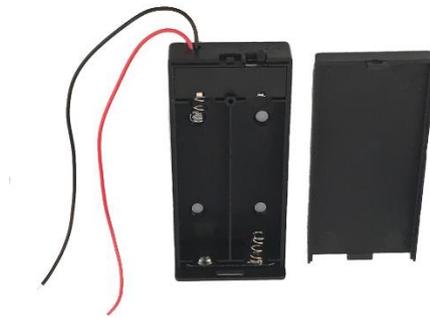
O kit possui um chassi superior, cujas dimensões aproximadas são de 225x150x3 milímetros, e outro inferior, com dimensões de aproximadamente 260x150x3 milímetros. Eles são importantes para a sustentação e resistência do carrinho, uma vez que em cima deles são parafusados todos os outros componentes. Além disso, o sistema com dois andares distribui o peso e o espaço ocupado pelos componentes instalados.

O modelo fornecido no kit, pelo fabricante, é o de chassis para veículos 4WD, sigla que significa *4 wheel drive*, ou tração nas quatro rodas em português, e é usada pelos fabricantes para denominar carrinhos que possuem 4 rodas motrizes, ou seja, 4 rodas capazes de tracionar.



*Figura 2: Chassi semelhante ao fornecido pelo fabricante.
Fonte: <https://www.Arduinolandia.com.br>.*

Para energizar o Arduino e os outros componentes do carrinho, são usadas duas baterias instaladas em um suporte plástico fixo no chassi superior. As baterias recomendadas são de lítio-íon, um metal leve, com grande potencial eletroquímico. Elas possuem capacidade de corrente de 2A, Tensão nominal de 3.7V e são caracterizadas pelo número 18650, cujos primeiros dois dígitos indicam o diâmetro em milímetros e o restante descreve o comprimento da pilha, também em milímetros, seguido do dígito zero. Essa bateria apesar de grande, se comparada com as pilhas do formato AA e AAA, é adequada para aplicações portáteis como o veículo em questão.



*Figura 3: Suporte para baterias.
Fonte: <http://osoyoo.com/>.*

O Arduino não possui fonte própria de energia, ele precisa ser alimentado por uma fonte externa. Para não causar danos ao microcontrolador, a placa trabalha com uma tensão de até 5V, limite que pode ser ampliado usando o regulador de tensão da própria placa. Essa alimentação, conforme indicado nas Figuras 4 e 5 pode ser feita por meio do conector de entrada, cuja saída deve ser em corrente contínua (DC), pelos pinos destinados a alimentação contidos na própria placa ou via USB com ajuda do cabo para transferência de dados da placa para o computador e do computador para a placa.



Figura 4: Pinos de alimentação e conector de entrada do Arduino.
Fonte: Adaptado de <http://www.eduino.com.br>.

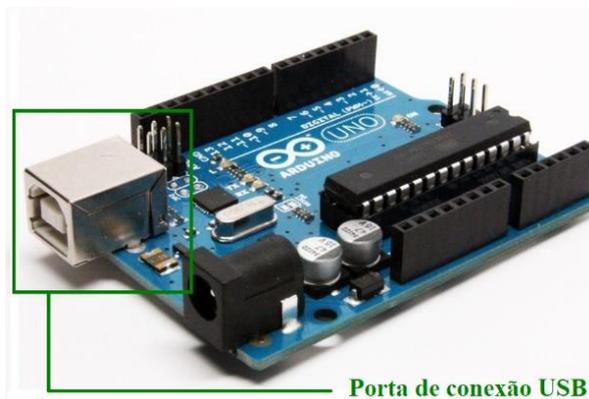


Figura 5: Conexão USB do Arduino.
Adaptado de <https://www.theengineeringprojects.com>

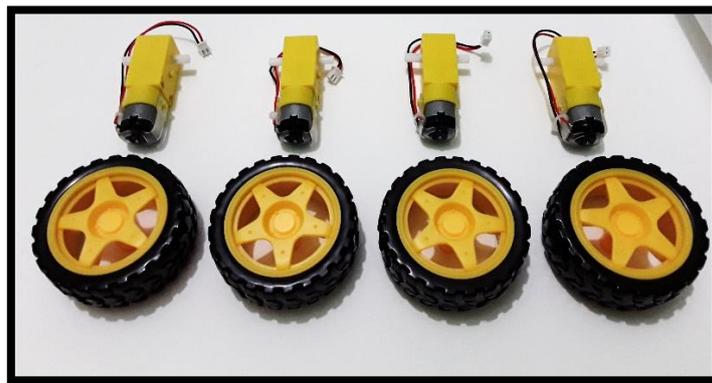
O kit possui um conector de alimentação DC. Então na montagem e utilização do veículo deste kit a placa Arduino e a ponte H, que será apresentada posteriormente, serão alimentadas por meio do conector e das baterias.



Figura 6: Conector de alimentação DC.
Fonte: <http://osoyoo.com/>.

O kit *Robot Car Starter* também contém quatro rodas de plástico cuja área de contato com a superfície é feita em borracha e quatro motores DC com caixa de redução e eixo duplo. Estes motores, responsáveis pela movimentação das rodas, contém um conjunto de engrenagens para reduzir sua velocidade de rotação e aumentar o torque. A velocidade é reduzida e o torque é aumentado na proporção dada pela relação entre os diâmetros do eixo do motor e da roda que, segundo o fabricante, o carrinho usado possui redução de 1:48, permitindo que veículo suporte o peso dos componentes e não realize movimentos bruscos.

As rodas possuem aproximadamente um diâmetro de 65 milímetros e 25 milímetros de espessura e de acordo com as informações fornecidas pelo fabricante, a tensão de alimentação dos motores é de 3 a 6V em DC, o torque de 0,35 a 0,80 Kgf/cm, a rotação do motor e a velocidade do veículo a 6V em DC são de 260 rpm e 1 m/s respectivamente.



*Figura 7: Rodas e motores com caixa de redução e eixo duplo.
Fonte: Fotografia do kit utilizado.*

Para a montagem e as conexões entre os componentes eletrônicos, o fabricante disponibiliza no kit diversos tipos de *jumpers*, parafusos, porcas e pilares de cobre conforme mostra a Figura 8.

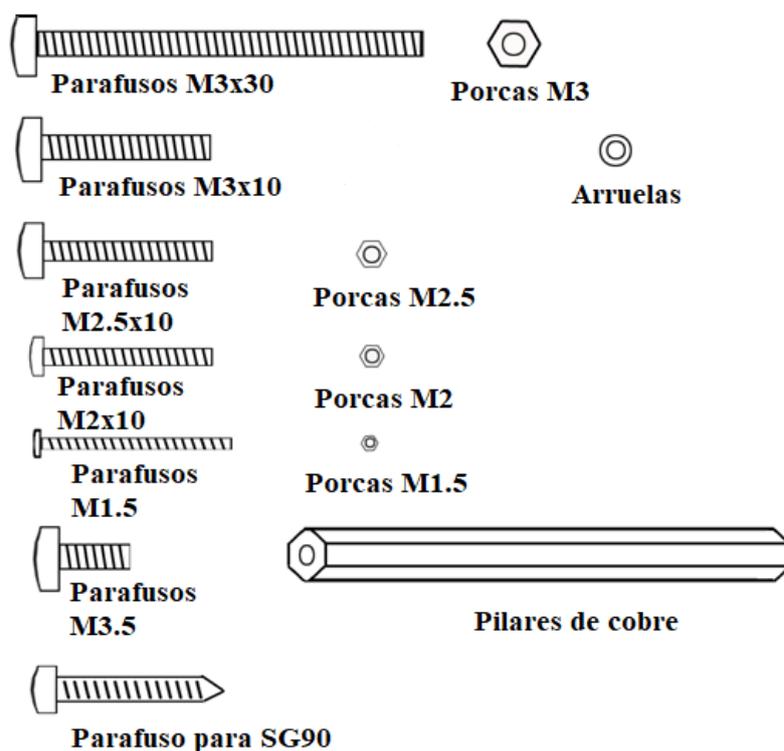


Figura 8: Lista de parafusos e porcas utilizadas.
 Fonte: Adaptação de <http://www.osoyoo.com/>.

Dos componentes que integram o kit, o que possibilita que o veículo se movimente através da navegação ótica, é o módulo de sensor ótico no modelo TCTR5000. Para que a movimentação se concretize, no kit *Robot Car Starter*, são necessários cinco sensores deste tipo, eles possuem saída de transistor, luz indicadora e ajuste de sensibilidade por meio de um potenciômetro. Esses sensores, de acordo com seu *Datasheet*, documento que traz todas as especificações de um componente eletrônico, são compostos por um LED que emite um sinal infravermelho, cuja frequência não é perceptível a seres humanos, esse sinal é refletido pela superfície e captado por um fototransistor que filtra a luz natural e identifica a cor preta ou branca. Além disso, em harmonia com os dados oferecidos pelo fabricante, esse sensor tem dimensões de 38 x 14 x 18 mm, operando com uma tensão de 3,3-5V em DC e é capaz de captar a faixa a uma distância de até 25 milímetros, operando melhor a uma distância de 0,2 a 15 milímetros.

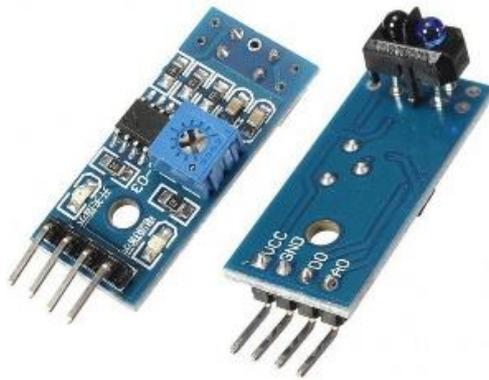


Figura 9: Sensor fornecido pelo fabricante.
FONTE: <http://www.huinfinito.com.br>

O módulo *driver OSOYOO Model-X*, segundo o fabricante, é uma versão melhorada do convencional módulo ponte H L298N. As alterações foram feitas visando facilitar a montagem e estabilidade das conexões. Assim como o módulo padrão, este possui chaves organizadas formando um “H”, que controlam a potência, a velocidade e a direção de motores de corrente contínua (DC) por meio de sinais elétricos.

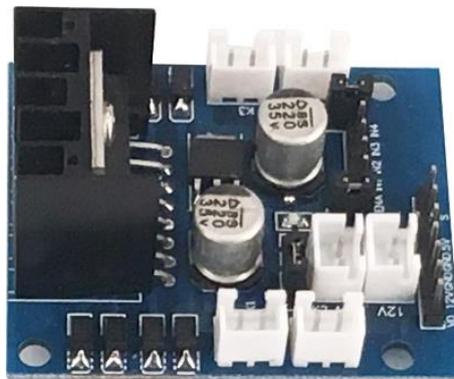


Figura 10: Módulo fornecido pelo fabricante.
Fonte: <http://osoyoo.com/>.

Para viabilizar a movimentação do carro robótico, o kit escolhido usa uma plataforma aberta de prototipagem eletrônica chamada Arduino. Para McRoberts

“Em termos práticos, um Arduino é um pequeno computador que você pode programar para processar entradas e saídas entre o dispositivo e os componentes externos conectados a ele.” (McRoberts, 2011, p.22)

O Arduino surgiu em 2005 no norte da Itália, a partir da necessidade que um professor tinha em utilizar ferramentas com baixo custo e fácil manipulação para que os alunos do *Interaction Design Institute* trabalhassem com a tecnologia e a programação em projetos de arte, automação e robótica.



Figura 11: Arduino UNO R3.
FONTE: <http://osoyoo.com/>.

De acordo com a página oficial do Arduino na internet, a plataforma teve cinco criadores, Massimo Banzi, David Cuartielles, David Mellis, Gianluca Martino e Tom Igoe, cada um deles contribuiu com seus conhecimentos para projetar, produzir e divulgar o Arduino. Os professores, de design, Massimo Banzi e de programação, David Cuartielles trabalharam no projeto inicial do Arduino, enquanto o estudante David Mellis colaborou como desenvolvimento do software, o professor do curso de Mestrado em Telecomunicações Interativas (ITP) da Universidade de Nova York (NYU), Tom Igoe, participou da equipe fundadora por adotar e orientar o projeto Arduino, e por fim, o engenheiro Gianluca Martino, contribuiu com o design do hardware e providenciou a fabricação dos primeiros protótipos.

A utilização do Arduino em projetos variados se difundiu rapidamente por todo o mundo principalmente em virtude do fato de ser um software livre, o que, de acordo com Castell (2005), são programas que

[...] “são entregues à comunidade com o código fonte aberto e disponível, permitindo que a ideia original possa ser aperfeiçoada e devolvida novamente a comunidade.” (Castell, 2005, p. 230)

Em outras palavras, um software livre, como o do Arduino, é um programa que permite que o usuário utilize, modifique, copie e distribua livremente. Essa liberdade só é possível para softwares *open source*, que são aqueles softwares cujo código fonte é aberto, por isso a principal característica de um software livre é o fato dele disponibilizar o código que lhe dá origem aos seus usuários, o que permite que o software seja aprimorado e adaptado às necessidades de quem vai utilizá-lo.

Outro motivo pelo qual a popularidade do Arduino cresceu rapidamente foi o baixo custo, uma vez que seus criadores almejavam uma plataforma que custasse menos do que o valor gasto pelo estudante quando saísse para comer uma pizza. E de fato, até hoje o Arduino é considerado acessível justamente porque é barato e fácil de usar. Alves *et al* (2012), argumenta que o Arduino, devido a essas vantagens, é uma das tecnologias usadas na Robótica Educacional de Baixo Custo (REBC), ele diz:

[...] “A REBC utiliza materiais alternativos, recursos de hardware e de software livre, tais como o projeto Arduino, como forma de se viabilizar projetos educacionais na área de RE.” (ALVES *et al.* 2012, p.164)

Outro aspecto de extrema importância sobre o Arduino é o microcontrolador que ele possui. Oliveira (2006) descreve um microcontrolador em poucas palavras dizendo que ele é um circuito integrado disponível nos mais variados tipos de encapsulamento e destinados ao tratamento de sinais digitais, ou seja, o microcontrolador é quase um computador que pode realizar diversas funções desde que seja programado. Os microcontroladores como o contido no Arduino UNO R3 que compõe o kit usado neste trabalho, podem ser programados e tem a capacidade de realizar novas tarefas à medida que é reprogramado.

O Arduino UNO R3 contém um microcontrolador no modelo ATmega328, que é programado usando a linguagem de programação C, que é considerada uma linguagem de alto nível.

A Figura 12 mostra o *layout* do Arduino UNO R3, neste layout observam-se seis pinos analógicos e catorze pinos digitais, que podem ser definidas tanto como

entrada como saída, e ainda alguns desses pinos digitais podem ser programados para fornecerem uma saída de modulação por largura de pulso, *Pulse Width Modulation* (PWM).

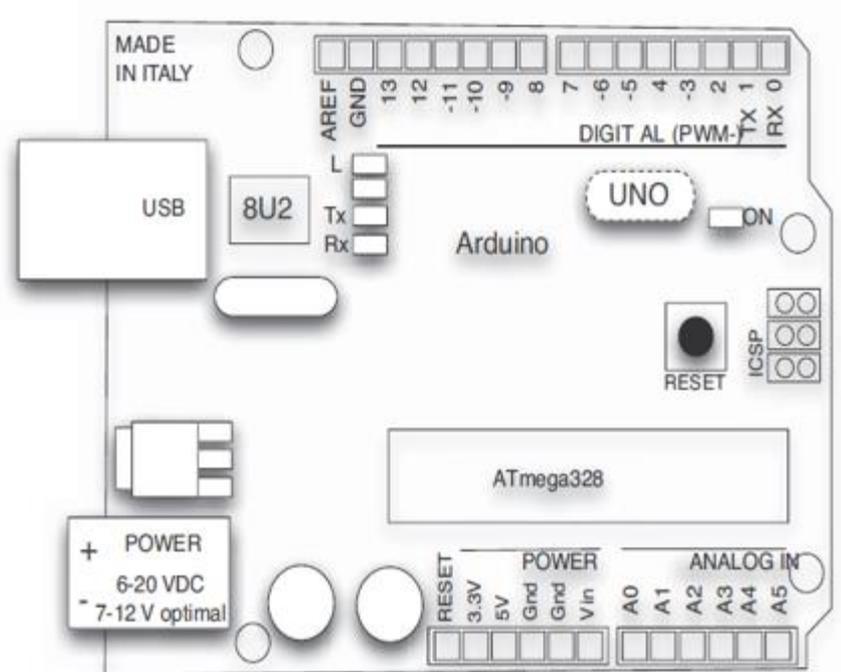


Figura 12: Layout da placa e pinos do Arduino Uno R3.
Fonte: Arduino em ação (2013).

Para o estudo da movimentação de um AGV, escolhemos um kit que usa o Arduino para controlar a navegação ótica pelos motivos já citados, além disso, o Arduino possui hardware e softwares livres com um preço que acessível à maioria das escolas sendo de fácil manipulação até mesmo para pessoas com pouco ou nenhum conhecimento de eletrônica.

Além desses motivos, Cury e Hirschmann, *apud* Silva (2017), afirmam que o Arduino é uma ferramenta construtivista na educação matemática e diz que atividades com ele estimula a socialização em grupo, a criatividade, o raciocínio lógico e estimula a pesquisa.

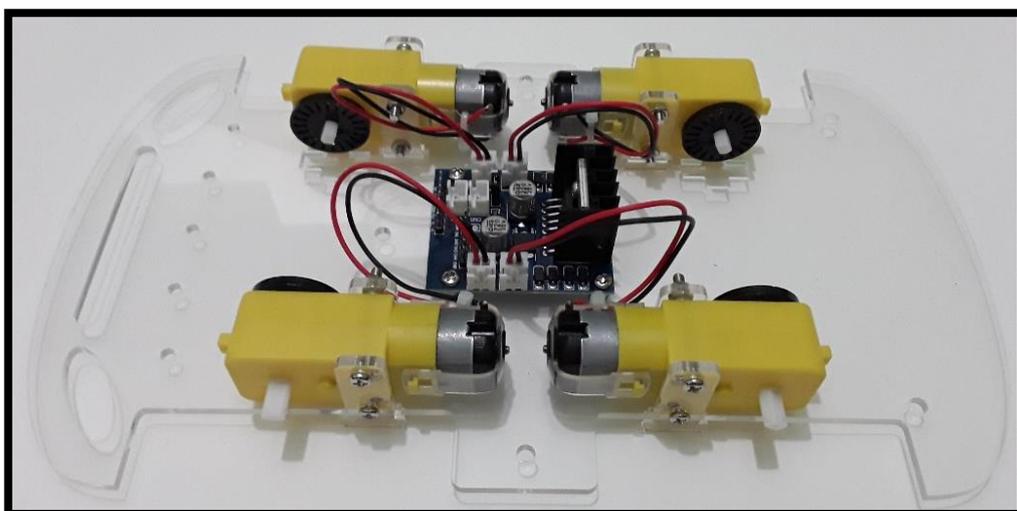
2.1.2. Montagem

Podemos consultar um tutorial detalhado para montagem tanto no CD que compõe o kit quanto no site <http://osoyoo.com/category/Arduinoraspikit/smart-car->

kit/. Este tutorial está separado em cinco lições que trazem desde tarefas básicas que auxiliam na montagem da estrutura mecânica do veículo e que realizam movimentos simples, tarefas que trabalham com a utilização de sensores infravermelho e controle remoto para determinar o trajeto da movimentação, até tarefas que trabalham a instalação de sensores seguidor de linha que permitem que o veículo se movimente através do reconhecimento de uma linha preta e do contraste dessa linha com fundo branco.

A primeira lição disponibilizada pelo fabricante tem por objetivo montar a estrutura mecânica do veículo e programar o carro para realizar movimentos simples. De acordo com esta lição, devemos utilizar os parafusos e porcas M3x30 para parafusar os quatro motores no chassi inferior. Com os parafusos M3x10 devemos instalar o módulo driver do Motor no chassi inferior, e feito isso, conectar os módulos k1, k2, k3 e k4 aos motores, como mostra a Figura 13. Após isso, no chassi superior, podemos instalar o Arduino UNO, o suporte para bateria e o medidor de tensão no chassi superior usando os parafusos e porcas M3x30, como apresentado na Figura 14.

O próximo passo é fazer as conexões do suporte para a bateria ao Arduino e do medidor de tensão e do Arduino ao módulo do driver do motor conforme indicam os diagramas nas Figuras 15,16 e 17. Concluindo isso, podemos conectar o chassi superior e inferior usando os pilares de cobre e os parafusos e por fim instalar as rodas, Figura 18.



*Figura 13: Instalação dos motores, driver do motor e conexões.
Fonte: Fotografia da montagem do kit.*

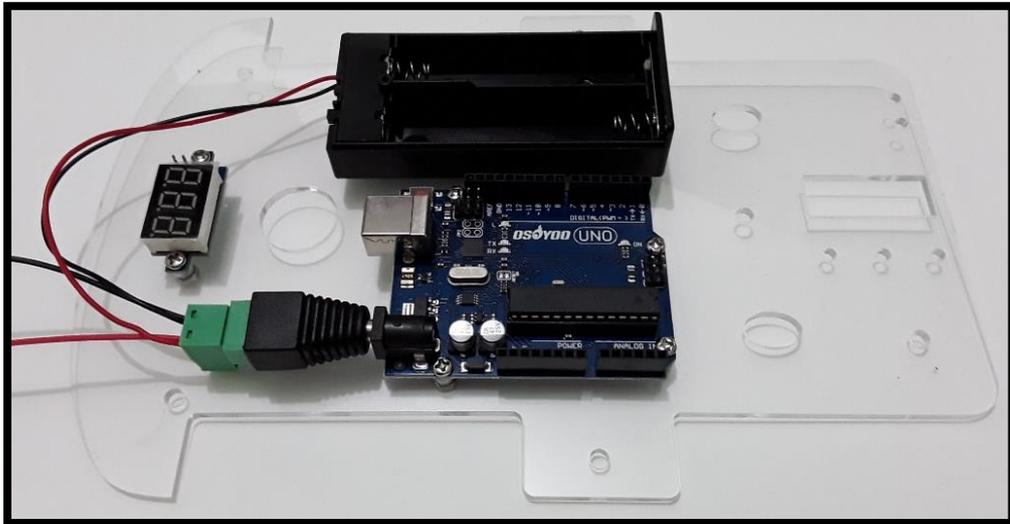


Figura 14: Instalação do Arduino e sua fonte de alimentação, do suporte para bateria e do medidor de voltagem.
Fonte: Fotografia da montagem do kit.

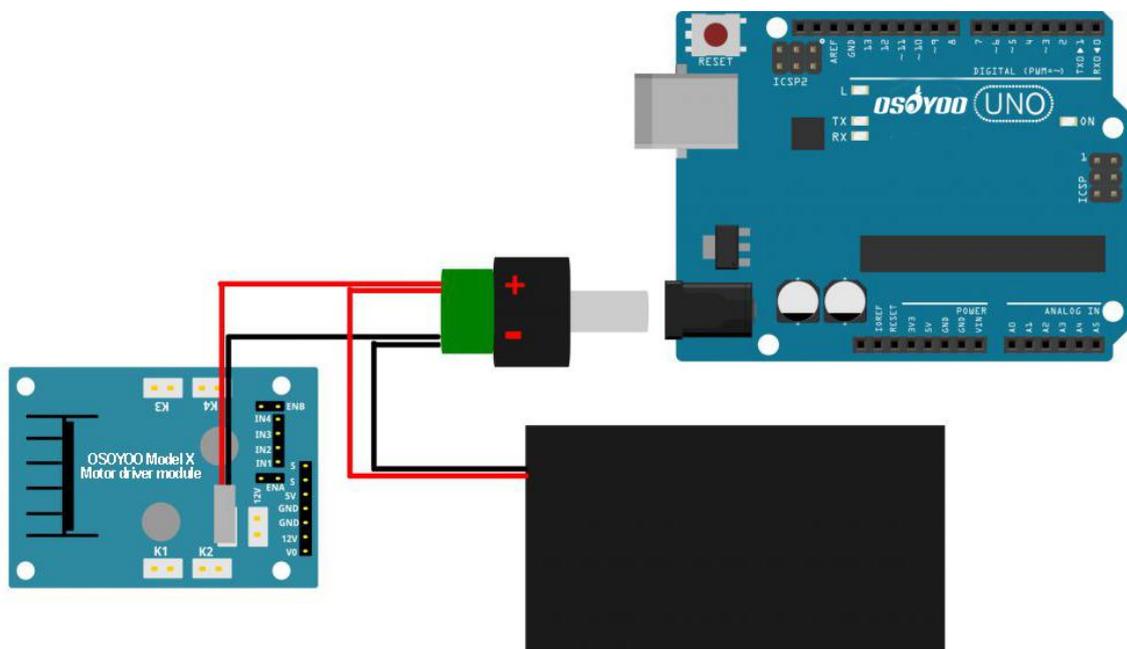


Figura 15: Conexões entre o suporte da bateria, o Arduino e o driver do motor.
Fonte: <http://osoyoo.com/>.

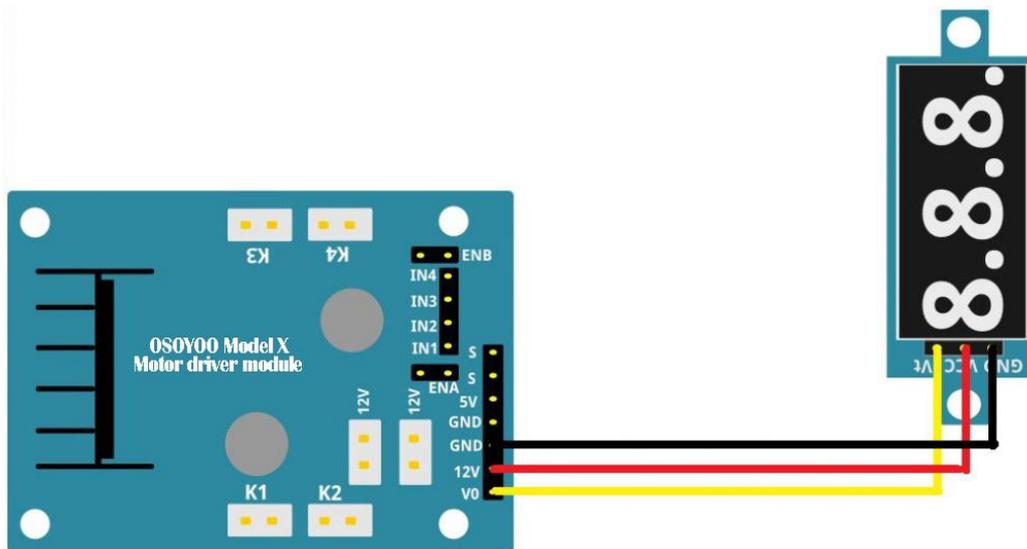


Figura 16: Conexões entre o driver do motor e o medidor de voltagem.
 Fonte: <http://osoyoo.com/>.

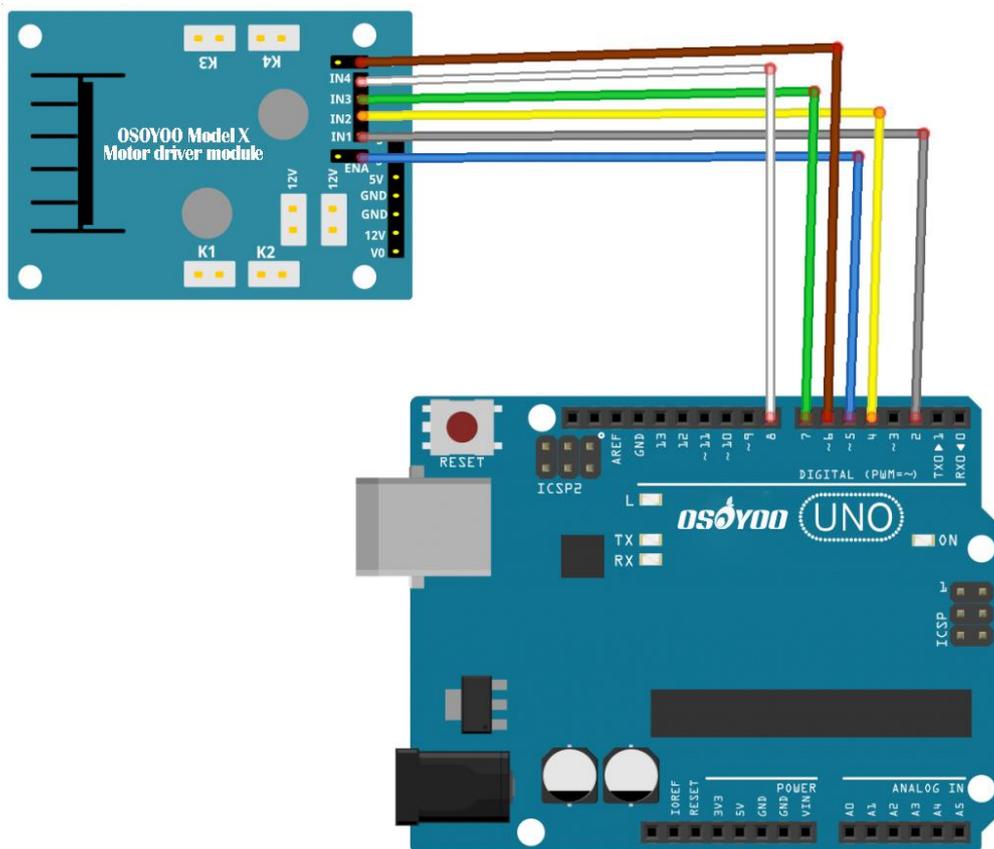


Figura 17: Conexões entre o Arduino e o driver do motor.
 Fonte: <http://osoyoo.com/>.

Com a primeira lição finalizada o veículo deve ficar conforme ilustrado na Figura 18.

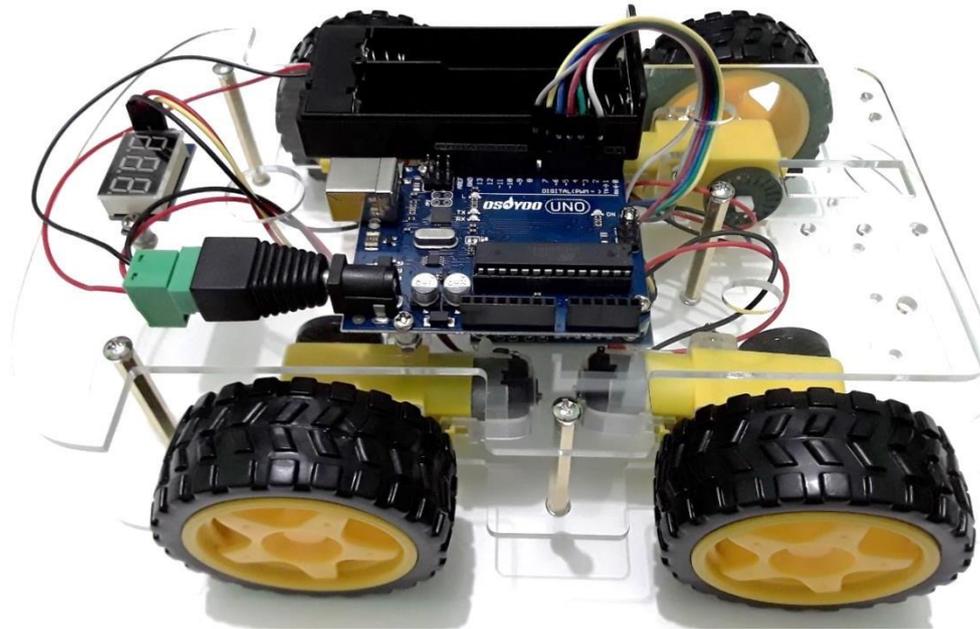


Figura 18: Fotografia do veículo após a finalização da primeira lição.
Fonte: Fotografia da montagem do veículo.

Na quarta lição, são instalados os cinco sensores de rastreamento, ou seja, os sensores seguidores de linha, a partir do que já foi realizado na primeira lição. Com os parafusos M3x10, devemos instalar os sensores de rastreamento na frente do chassi inferior e fazer as conexões conforme a Figura 20.

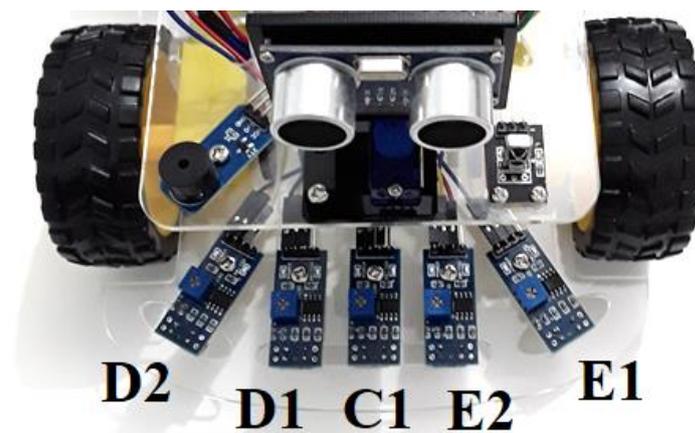


Figura 19: Instalação dos sensores.
Fonte: Fotografia da montagem do kit.

Tabela 1: Conexões entre o Arduino e os sensores.

Pinos dos sensores	Entradas do Arduino
VCC	5V
GND	GND
(E1) DO	(D) 3
(E2) DO	(A) 1
(C1) DO	(A) 2
(D1) DO	(A) 3
(D2) DO	(D) 10

Fonte: Adaptação de <http://osoyoo.com/>.

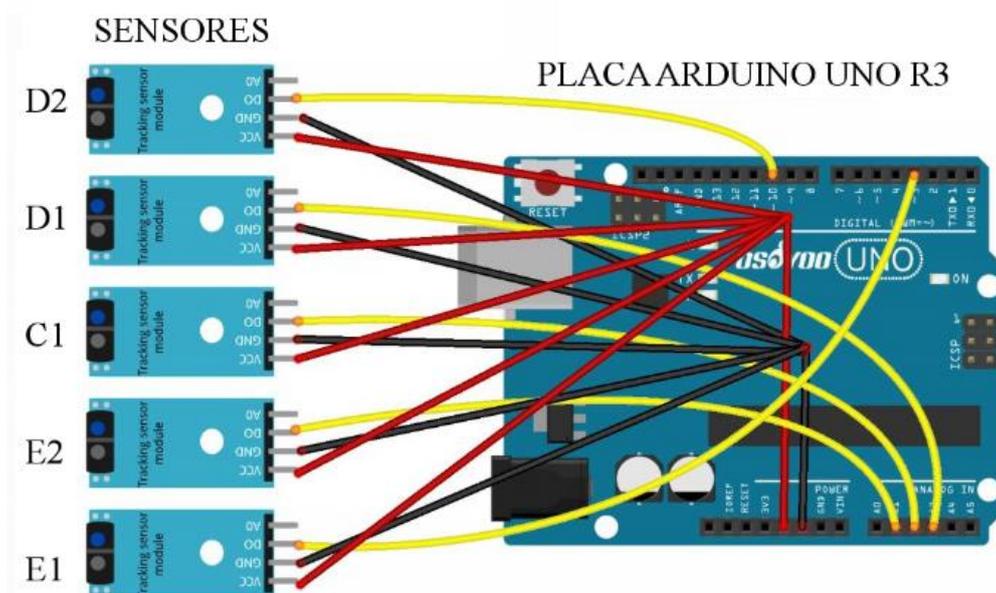


Figura 20: Instalação dos sensores de rastreamento e conexões entre o Arduino e os sensores.

Fonte: <http://osoyoo.com/>.

Após realizar a montagem necessária, para finalizar a primeira lição, ainda devemos instalar o software de programação Arduino IDE e o código de amostra fornecido pelo fabricante, que podem ser baixados gratuitamente em <https://www.Arduino.cc> e <http://osoyoo.com/driver/smartcar-lesson4.zip> respectivamente. Feito isso, com o cabo USB podemos conectar o Arduino UNO ao PC e carregar o código de programação desejado no Arduino. Só então podemos inserir com cuidado as baterias.

3. ÁLGEBRA LINEAR

A Álgebra Linear é o ramo da Matemática que usa os vetores, os espaços vetoriais e as funções lineares, para estudar sistemas de equações lineares algébricas e diferenciais. Para isso, ela também conta com conceitos, que incluem as matrizes, os sistemas lineares e os determinantes, sendo amplamente aplicada em diferentes áreas do conhecimento e de acordo com Lipschutz (1972), tornou-se essencial para a Física, Engenharia, Ciência da Computação, Economia e Estatística.

Este capítulo aborda a teoria matemática envolvida no estudo das matrizes e transformações lineares no plano, tomando como base as obras de Boldrini (1980), Calioli (1993), Lay (1999), Lipschutz (1972), Anton e Rorres (2012) e Steinbruch e Winterle (1987).

3.1 Espaço vetorial

Considere um conjunto V não vazio, que tem definidas as operações de adição entre elementos desse conjunto, $u + v \in V$ para $\forall u, v \in V$, e multiplicação por escalar $\alpha u \in V$ para $u \in V$ e $\alpha \in R$.

Chamamos V de um espaço vetorial real se forem válidos os seguintes axiomas $\forall u, v, w \in V$ e $\forall \alpha, \beta \in R$

$$A1) u + (v + w) = (u + v) + w$$

$$A2) u + v = v + u$$

$$A3) \exists 0 \in V \text{ tal que } \forall v \in V, \quad v + 0 = v$$

$$A4) \exists (-v) \in V \text{ tal que } \forall v \in V, \quad v + (-v) = 0$$

$$M1) \alpha(\beta u) = (\alpha\beta)u$$

$$M2) (\alpha + \beta)u = \alpha u + \beta u$$

$$M3) (\alpha)(u + v) = \alpha u + \alpha v$$

$$M4) 1(u) = u$$

Partindo dos conceitos de Espaços vetoriais, vamos definir transformações e transformações lineares para posteriormente aplicá-las no plano cartesiano.

3.2 TRANSFORMAÇÕES

A transformação, $f: A \rightarrow B$, também denominada como aplicação, transformada ou função, apesar de função ser um termo amplamente usado no meio acadêmico para se referir a aplicações de valores reais ou complexos, é uma relação entre dois conjuntos não vazios A e B , em que cada elemento $x \in A$ está associado a um único elemento $y \in B$. Além disso, o conjunto A recebe o nome domínio enquanto que o B , contradomínio.

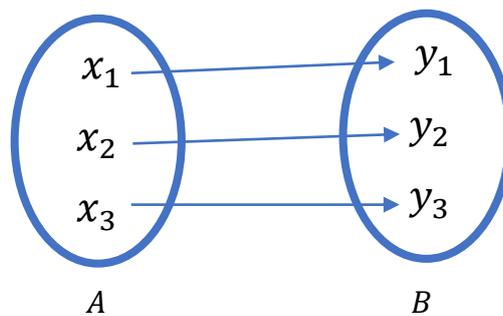


Figura 21: Diagrama de Venn.

Fonte: Diagrama de Venn elaborada com a ferramenta para inserir formas no Microsoft Word.

A Figura 21 mostra a transformação por meio de flechas que levam elementos do domínio à sua imagem no contradomínio. Iezzi (2013) argumenta que uma relação não é uma aplicação ou função se não satisfizer alguma das condições a seguir:

“1ª) É necessário que todo elemento $x \in A$ participe de pelo menos um par $(x, y) \in f$, isto é todo elemento de A deve servir como ponto de partida de flecha. 2ª) É necessário que cada elemento $x \in A$ participe de apenas um único par $(x, y) \in f$, isto é, cada elemento de A deve servir como ponto de partida de uma única flecha.” (Iezzi, 2013, p.81)

De acordo com as condições citadas acima a Figura 21 representa uma transformação porque um único elemento do domínio não possui duas imagens

distintas e não existe elemento do domínio que não seja associado a nenhuma imagem no contradomínio. Partindo disso, as figuras abaixo trazem mais exemplos de relações entre dois conjuntos, a Figura 22 mostra exemplos de funções enquanto que a Figura 23 mostra algumas relações que não respeitam as condições acima e, portanto, não são consideradas funções.

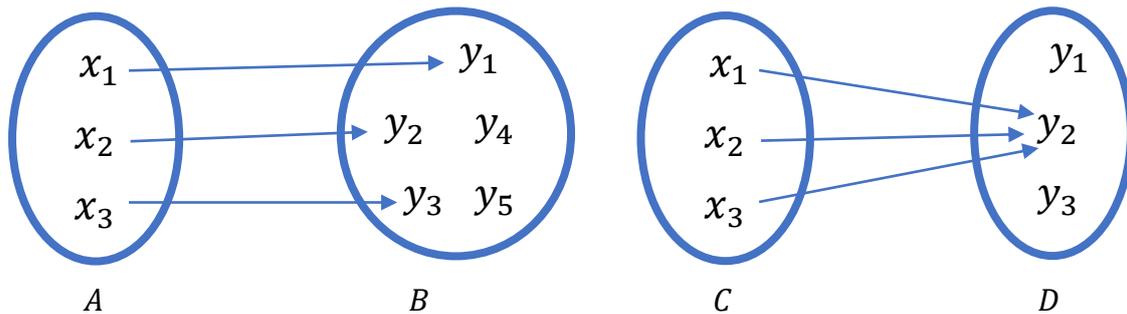


Figura 22: Outros exemplos de funções ou transformações.
 Fonte: Diagrama de Venn elaborada com a ferramenta para inserir formas no Microsoft Word.

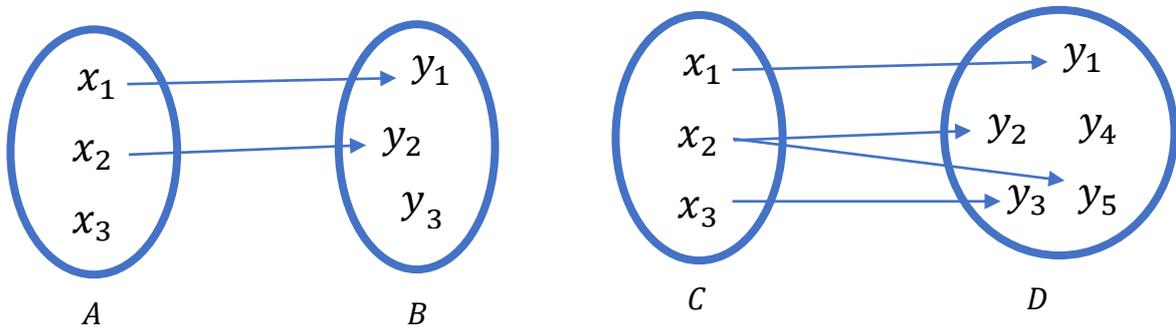


Figura 23: Exemplos de relação que não são funções ou transformações.
 Fonte: Diagrama de Venn elaborada com a ferramenta para inserir formas no Microsoft Word.

As transformações citadas anteriormente podem ser aplicadas aos espaços vetoriais definidos no começo do capítulo. Então, Dados dois espaços vetoriais V e W , dizemos que $T: V \rightarrow W$ é uma transformação ou uma função que associa a cada vetor $v \in V$ corresponde a um único vetor $w \in W$, ou seja, é uma regra que transforma elementos do espaço vetorial V em elementos do espaço vetorial W . Considerando que $v \in V$ tem um único vetor $w \in W$, denominado como imagem de v e indicado por $w = T(v)$.

3.2.2 Transformação Linear

Com base nos conceitos de transformações já explanados, uma transformação T é chamada uma transformação linear de V em W se, para quaisquer vetores u e v em V e um escalar c qualquer valem:

- i. $T(u + v) = T(u) + T(v)$
- ii. $T(cv) = cT(v)$

Em outras palavras, uma transformação linear é uma aplicação entre dois espaços vetoriais que preserva duas operações básicas, a soma de vetores e a multiplicação por escalar.

Tais transformações também podem ser realizadas por meio de matrizes por isso muitas vezes elas recebem o nome de transformações matriciais, usando o que Lay (1999) descreve como a ideia fundamental da álgebra linear, que é a capacidade de escrever uma combinação linear de vetores por meio de produto de matrizes.

3.3 TRANSFORMAÇÕES GEOMÉTRICAS NO PLANO

Dentre as transformações lineares existentes, contamos com um grupo específico de aplicações que podem ser interpretadas geometricamente e observadas no plano cartesiano e no espaço. Essas transformações conhecidas como Transformações Geométricas serão esplanadas ao longo deste capítulo. Elas possuem grande aplicabilidade em diversas áreas, incluindo a computação e a produção de Animações.

As Transformações Geométricas são empregadas quando, aumentamos ou diminuimos uma imagem com a ferramenta de zoom, observamos um reflexo em uma superfície espelhada, posicionamos melhor uma figura ou fotografia que está de ponta cabeça, e, além disso, Schwartz (2016) indica o uso do cisalhamento para produzir em uma imagem os "efeitos do tremor de terra sobre a estrutura de um edifício". E no contexto da Matemática as transformações ainda podem ser usadas para facilitar os cálculos com cônicas cujos eixos não são paralelos aos eixos cartesianos, por exemplo, uma elipse cujo eixo maior coincide com a reta $y = x$.

Antes de analisar as transformações geométricas, note que um par (x, y) no plano cartesiano, pode ser entendido como um ponto de coordenadas x e y ou como um vetor cujas componentes são x e y . Para não atrapalhar a visualização das transformações geométricas abaixo, usamos pontos para representar a extremidade de vetores que se iniciam na origem do plano cartesiano, como mostra a Figura 24. Observe que não se tratam de pontos, mas de vetores localizados num sistema de coordenadas.

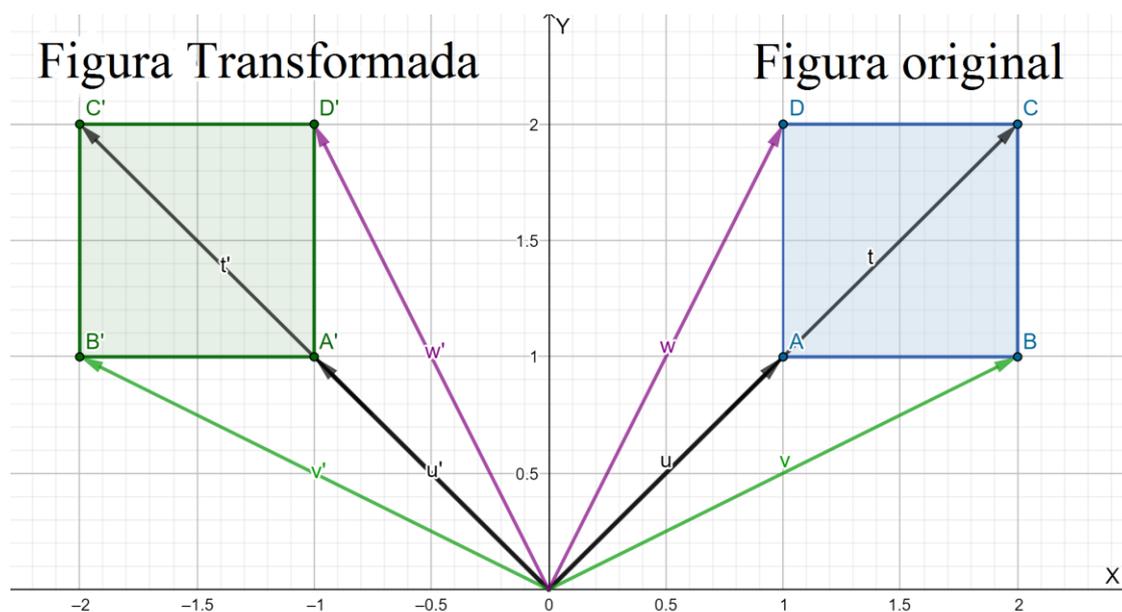


Figura 24: Quadrado formado pelos vetores \vec{u} , \vec{v} , \vec{t} e \vec{w} e quadrado formado pelos vetores transformados. Fonte: Elaborada no Geogebra.

3.2.1 Dilatação

A transformação linear que aumenta o valor do módulo de um vetor ou o tamanho de uma figura geométrica posicionada em um sistema de coordenadas é chamada dilatação.

Os objetos são aumentados na direção x com base no parâmetro α e, de acordo com β na direção y , de modo que $|\alpha| > 1$, $|\beta| > 1$ ou $|\alpha|$ e $|\beta| > 1$.

Tais restrições existem porque quando α , β ou ambos são iguais a 1, as dimensões da figura ou do vetor não sofrem alterações, pois o número 1 é o

elemento neutro da multiplicação. Além disso, consideramos o módulo dos parâmetros porque quando α , β ou ambos são negativos, temos a dilatação associada à outra transformação geométrica chamada de reflexão, e neste momento nosso foco é apenas na dilatação.

Essa transformação, $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, pode ser descrita algebricamente da seguinte forma: $T(x, y) = (\alpha x, \beta y)$ ou por meio da matriz:

$$T = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$$

As novas coordenadas x' e y' , da figura ou do vetor transformado podem ser representadas por:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha x \\ \beta y \end{pmatrix}$$

Considere abaixo um exemplo de dilatação cuja nova figura e o vetor obtido pela transformação possuem o dobro de algumas, ou de todas suas dimensões originais.

Analise primeiro o caso em que os vetores iniciais possuem as seguintes coordenadas $A = (0, 2)$ e $B = (2, 0)$ e os parâmetros $\alpha = 2$ e $\beta = 1$, ou seja, a transformação é dada por $T_1(x, y) = (2x, y)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 4 \\ 0 \end{pmatrix}$$

Após a dilatação os vetores são $A' = (0, 2)$ e $B' = (4, 0)$.

Observe agora a transformação dos mesmos vetores A e B , mas com os parâmetros $\alpha = 1$ e $\beta = 2$, algebricamente a transformação pode ser escrita da como $T_2(x, y) = (x, 2y)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$$

Logo, os vetores A e B transformados, são $A' = (0,4)$ e $B' = (2,0)$.

E por fim, ainda com os mesmos vetores podemos fazer outra transformação que altere as duas direções do objeto, com os seguintes parâmetros: $\alpha = \beta = 2$. Com isso, a transformação é dada por $T_3(x, y) = (2x, 2y)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 4 \\ 0 \end{pmatrix}$$

Os vetores A e B , depois da transformação possuem as seguintes coordenadas $A' = (0,4)$ e $B' = (4,0)$.

Os vetores iniciais $A = (0,2)$ e $B = (2,0)$ podem ser representados no plano cartesiano e essa disposição pode ser observada na Figura 25 que mostra um quadrado formado pelos vetores A e B .

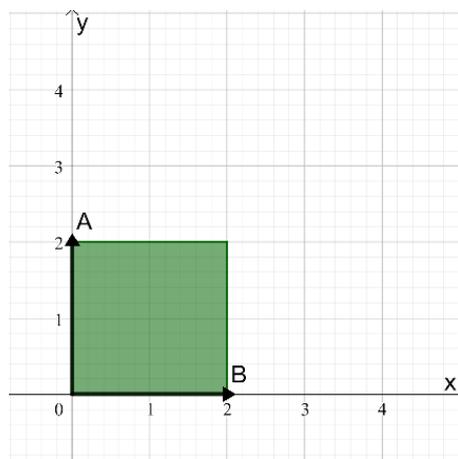


Figura 25: Quadrado formado pelos vetores A e B .
Fonte: Elaborada no Geogebra.

A Figura 26 mostra as transformações 1, 2 e 3 dos vetores A e B conforme calculado anteriormente, sendo possível observar como a dilatação de apenas uma ou nas duas coordenadas pode modificar a figura formada por meio dos vetores.

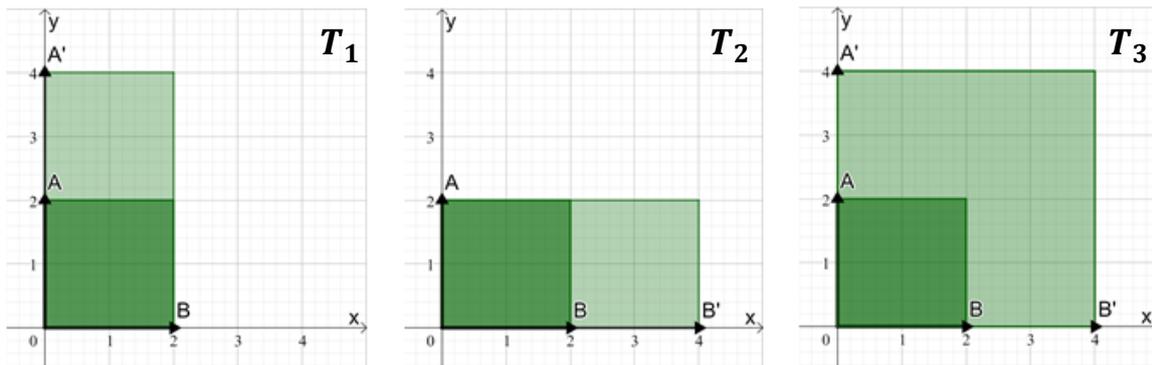


Figura 26: Figura elaborada no Geogebra para exemplificar as transformações T_1 , T_2 , e T_3 .
Fonte: Elaborada no Geogebra.

3.3.2. Contração

Assim como a dilatação modifica o tamanho de objetos aumentando suas dimensões parcial ou totalmente, a contração também é uma transformação linear que altera as medidas de um vetor ou de uma figura disposta no plano cartesiano, mas diminuindo tais dimensões. Para isto, os parâmetros obedecem à seguinte restrição: $0 < |\alpha| < 1$ e $0 < |\beta| < 1$.

A contração é uma transformação $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ e pode ser descrita como $T(x, y) = (\alpha x, \beta y)$. Neste caso, a matriz dessa transformação também é

$$T = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$$

As coordenadas x' e y' da nova figura ou do vetor transformado podem ser obtidas com a expressão abaixo:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha \cdot x \\ \beta \cdot y \end{pmatrix}$$

Observe agora alguns exemplos de contração em que a nova figura e o vetor obtido pela transformação possuem a metade de algumas, ou de todas as suas dimensões originais. Para isso, vamos considerar os mesmos vetores $A = (0,2)$ e $B = (2,0)$ usados nos exemplos anteriores.

Primeiramente, examine o caso cujos vetores são transformados sob os parâmetros $\alpha = \frac{1}{2}$ e $\beta = 1$, ou seja, a transformação é dada por $T_4(x, y) = \left(\frac{x}{2}, y\right)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Após a contração os vetores são $A' = (0,2)$ e $B' = (1,0)$.

Agora considere a transformação dos vetores A e B , com os parâmetros $\alpha = 1$ e $\beta = \frac{1}{2}$, algebricamente esta transformação pode ser escrita da seguinte forma $T_5(x, y) = (x, \frac{y}{2})$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$$

Logo, os vetores A e B transformados, são $A' = (0,1)$ e $B' = (2,0)$.

E por fim, vejamos ainda a transformação cujos parâmetros são $\alpha = \beta = \frac{1}{2}$.

Com isso a transformação é dada por $T_6(x, y) = (\frac{x}{2}, \frac{y}{2})$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

A transformação dos vetores A e B os associa aos vetores A' e B' que possuem as seguintes coordenadas $A' = (0,1)$ e $B' = (1,0)$.

Os vetores $A = (0,2)$ e $B = (2,0)$ são representados no plano cartesiano como na Figura 27 que mostra um quadrado formado pelos vetores A e B .

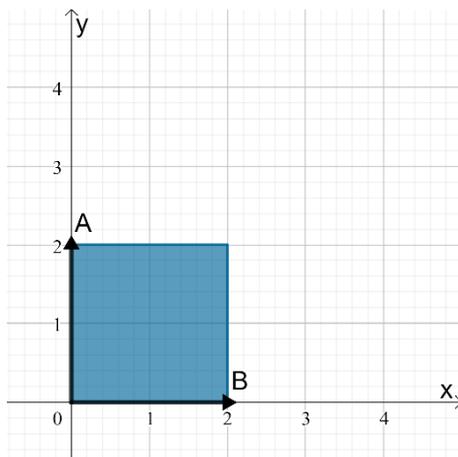


Figura 27: Quadrado formado pelos vetores A e B .
Fonte: Elaborada no Geogebra.

Os vetores A e B representados pela Figura 24 sofreram as transformações 4, 5 e 6 e estão retratados na Figura 28.

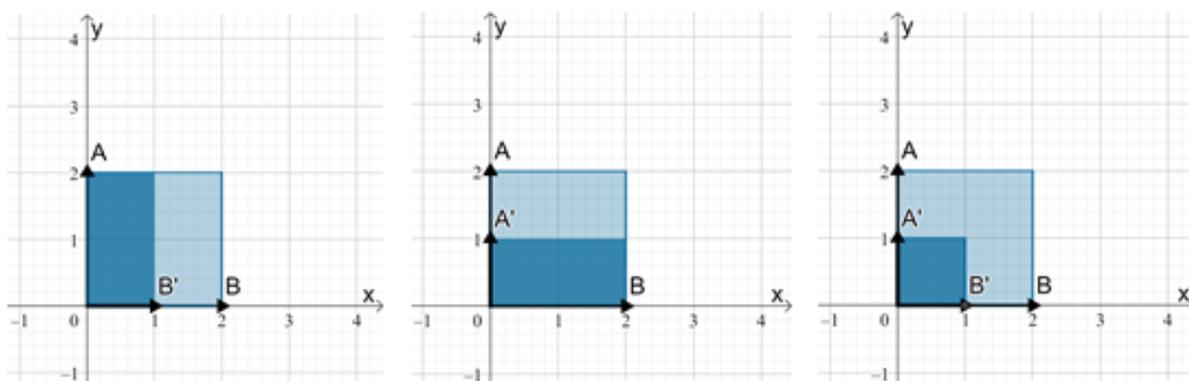


Figura 28: Figura elaborada no Geogebra para exemplificar as transformações 4, 5 e 6.
Fonte: Elaborada no Geogebra.

3.2.2 Rotação

A transformação geométrica que promove a rotação de um vetor ou de uma figura, em torno da origem do plano cartesiano, nos sentidos horário ou anti-horário é chamada de rotação.

Diferentemente da transformação de escala, citada anteriormente, que possui dois parâmetros, a transformação de rotação possui um único parâmetro θ que é determinado pelo ângulo formado entre o eixo das abscissas e a reta que passa pela origem do sistema de coordenadas e o ponto transformado.

Este parâmetro, quando positivo, implica diretamente no sentido da rotação, que será realizada no sentido anti-horário, enquanto que, caso negativo, a rotação acontecerá no sentido horário.

Tendo em vista o ponto pelo qual a rotação será realizada, a transformação $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ é escrita de forma algébrica como $T(x, y) = (x \cdot \cos(\theta) - y \cdot \text{sen}(\theta), x \cdot \text{sen}(\theta) + y \cdot \cos(\theta))$, sendo que $0 \leq \theta \leq 2\pi$. Essa transformação também pode ser expressa pela seguinte matriz:

$$T = \begin{bmatrix} \cos(\theta) & -\text{sen}(\theta) \\ \text{sen}(\theta) & \cos(\theta) \end{bmatrix}$$

Logo, a matriz que descreve a rotação é dada por:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\text{sen}(\theta) \\ \text{sen}(\theta) & \cos(\theta) \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x \cdot \cos(\theta) - y \cdot \text{sen}(\theta) \\ x \cdot \text{sen}(\theta) + y \cdot \cos(\theta) \end{pmatrix}$$

Consideremos alguns exemplos de rotações aplicadas sobre um quadrado que inicialmente está posicionado de forma que seus vértices sejam $A = (3,3)$, $B = (2,3)$, $C = (2,2)$ e $D = (3,2)$ como mostra a Figura 29. Lembrando novamente, que neste contexto, os vértices A, B, C e D do quadrado são as extremidades dos vetores que têm início na origem do plano cartesiano e terminam nos pontos A, B, C e D respectivamente.

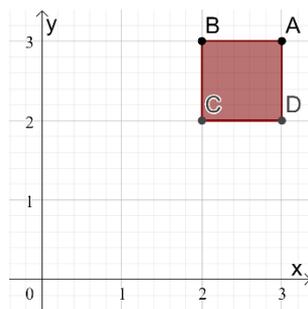


Figura 29: Quadrado que será rotacionado.
Fonte: Elaborado no Geogebra.

Esta transformação é dada por: $T_7(x, y) = \left(x \cdot \cos\left(\frac{\pi}{4}\right) - y \cdot \text{sen}\left(\frac{\pi}{4}\right), x \cdot \text{sen}\left(\frac{\pi}{4}\right) + y \cdot \cos\left(\frac{\pi}{4}\right) \right)$ e, será aplicada em cada um dos vértices do quadrado utilizando o parâmetro $\theta = \frac{\pi}{4}$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{4}\right) & -\text{sen}\left(\frac{\pi}{4}\right) \\ \text{sen}\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} \frac{3\sqrt{2} - 3\sqrt{2}}{2} \\ \frac{3\sqrt{2} + 3\sqrt{2}}{2} \end{pmatrix} = \begin{pmatrix} 0 \\ 3\sqrt{2} \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{4}\right) & -\text{sen}\left(\frac{\pi}{4}\right) \\ \text{sen}\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} \frac{2\sqrt{2} - 3\sqrt{2}}{2} \\ \frac{2\sqrt{2} + 3\sqrt{2}}{2} \end{pmatrix} = \begin{pmatrix} -\frac{\sqrt{2}}{2} \\ \frac{5\sqrt{2}}{2} \end{pmatrix}$$

$$\begin{pmatrix} x'_C \\ y'_C \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{4}\right) & -\text{sen}\left(\frac{\pi}{4}\right) \\ \text{sen}\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} \frac{2\sqrt{2} - 2\sqrt{2}}{2} \\ \frac{2\sqrt{2} + 2\sqrt{2}}{2} \end{pmatrix} = \begin{pmatrix} 0 \\ 2\sqrt{2} \end{pmatrix}$$

$$\begin{pmatrix} x'_D \\ y'_D \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{4}\right) & -\text{sen}\left(\frac{\pi}{4}\right) \\ \text{sen}\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} \frac{3\sqrt{2} - 2\sqrt{2}}{2} \\ \frac{3\sqrt{2} + 2\sqrt{2}}{2} \end{pmatrix} = \begin{pmatrix} \frac{\sqrt{2}}{2} \\ \frac{5\sqrt{2}}{2} \end{pmatrix}$$

A Figura 30 mostra a transformação T_7 .

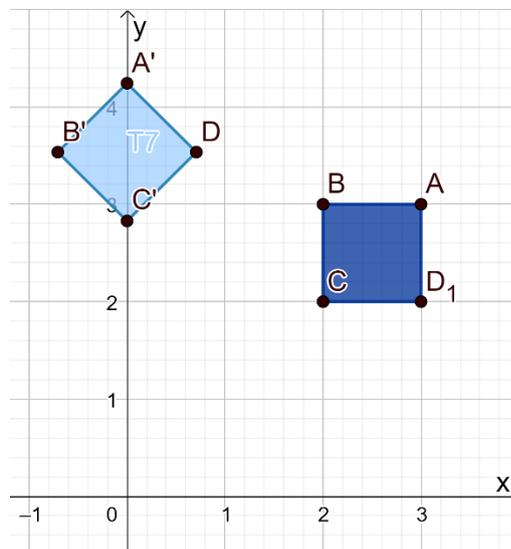


Figura 30: Imagem da transformação dos vetores cujas extremidades são A, B, C e D.
Fonte: Elaborado no Geogebra.

Note que, apesar da transformação T_7 ter sido aplicada apenas aos vértices do quadrado do exemplo anterior para facilitar a visualização, esta deveria ter sido aplicada a TODOS os pontos dos lados que compõem o quadrado. Porém, se

considerarmos que a figura é rígida ao transformar um ponto, transformamos com ele todos os pontos que pertencem a figura. Por esta razão, as próximas transformações serão aplicadas somente ao vértice $A = (3,3)$.

O quadrado inicial será alvo de outra rotação no sentido anti-horário cujo parâmetro será $\theta = \frac{\pi}{2}$, logo $T_8(x, y) = \left(x \cdot \cos\left(\frac{\pi}{2}\right) - y \cdot \text{sen}\left(\frac{\pi}{2}\right), x \cdot \text{sen}\left(\frac{\pi}{2}\right) + y \cdot \cos\left(\frac{\pi}{2}\right) \right)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{2}\right) & -\text{sen}\left(\frac{\pi}{2}\right) \\ \text{sen}\left(\frac{\pi}{2}\right) & \cos\left(\frac{\pi}{2}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} -3 \\ 3 \end{pmatrix}$$

A próxima transformação, ainda sob o quadrado ABCD, será $T_9(x, y) = \left(x \cdot \cos\left(\frac{3\pi}{4}\right) - y \cdot \text{sen}\left(\frac{3\pi}{4}\right), x \cdot \text{sen}\left(\frac{3\pi}{4}\right) + y \cdot \cos\left(\frac{3\pi}{4}\right) \right)$, ou seja, $\theta = \frac{3\pi}{4}$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{3\pi}{4}\right) & -\text{sen}\left(\frac{3\pi}{4}\right) \\ \text{sen}\left(\frac{3\pi}{4}\right) & \cos\left(\frac{3\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} \frac{-3\sqrt{2} - 3\sqrt{2}}{2} \\ \frac{3\sqrt{2} - 3\sqrt{2}}{2} \end{pmatrix} = \begin{pmatrix} -3\sqrt{2} \\ 0 \end{pmatrix}$$

As transformações a seguir serão rotações do mesmo vértice do quadrado utilizado nas transformações T_7, T_8 e T_9 , mas no sentido horário. A primeira transformação nesse sentido é $T_{10}(x, y) = \left(x \cdot \cos\left(-\frac{\pi}{4}\right) - y \cdot \text{sen}\left(-\frac{\pi}{4}\right), x \cdot \text{sen}\left(-\frac{\pi}{4}\right) + y \cdot \cos\left(-\frac{\pi}{4}\right) \right)$, com $\theta = -\frac{\pi}{4}$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(-\frac{\pi}{4}\right) & -\text{sen}\left(-\frac{\pi}{4}\right) \\ \text{sen}\left(-\frac{\pi}{4}\right) & \cos\left(-\frac{\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} \frac{3\sqrt{2} + 3\sqrt{2}}{2} \\ \frac{-3\sqrt{2} + 3\sqrt{2}}{2} \end{pmatrix} = \begin{pmatrix} 3\sqrt{2} \\ 0 \end{pmatrix}$$

Note que na transformação acima, usamos um dos conceitos citados por lezzi (2013), sobre a paridade das funções trigonométricas, dizendo que “a função seno é uma função ímpar e que a função cosseno é uma função par”, porque a definição de função par e função ímpar justifica a seguinte passagem abaixo, que foi utilizada na transformação 10 e será usada nas transformações 11, 12 e 13:

$$\text{sen}(-x) = -\text{sen}(x), \forall x \in \mathbb{R}$$

$$\text{cos}(-x) = \text{cos}(x), \forall x \in \mathbb{R}$$

A transformação 11 conta com o parâmetro $\theta = -\frac{\pi}{2}$ e é descrita algébrica-mente por $T_{11}(x, y) = \left(x \cdot \cos\left(-\frac{\pi}{2}\right) - y \cdot \text{sen}\left(-\frac{\pi}{2}\right), x \cdot \text{sen}\left(-\frac{\pi}{2}\right) + y \cdot \cos\left(-\frac{\pi}{2}\right)\right)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(-\frac{\pi}{2}\right) & -\text{sen}\left(-\frac{\pi}{2}\right) \\ \text{sen}\left(-\frac{\pi}{2}\right) & \cos\left(-\frac{\pi}{2}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{2}\right) & +\text{sen}\left(\frac{\pi}{2}\right) \\ -\text{sen}\left(\frac{\pi}{2}\right) & \cos\left(\frac{\pi}{2}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} 3 \\ -3 \end{pmatrix}$$

A rotação cujo parâmetro é $\theta = -\frac{3\pi}{4}$ é a transformação 12 que é expressa pela expressão $T_{12}(x, y) = \left(x \cdot \cos\left(-\frac{3\pi}{4}\right) - y \cdot \text{sen}\left(-\frac{3\pi}{4}\right), x \cdot \text{sen}\left(-\frac{3\pi}{4}\right) + y \cdot \cos\left(-\frac{3\pi}{4}\right)\right)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(-\frac{3\pi}{4}\right) & -\text{sen}\left(-\frac{3\pi}{4}\right) \\ \text{sen}\left(-\frac{3\pi}{4}\right) & \cos\left(-\frac{3\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{3\pi}{4}\right) & +\text{sen}\left(\frac{3\pi}{4}\right) \\ -\text{sen}\left(\frac{3\pi}{4}\right) & \cos\left(\frac{3\pi}{4}\right) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix}$$

$$\begin{pmatrix} \frac{-3\sqrt{2} + 3\sqrt{2}}{2} \\ \frac{-3\sqrt{2} - 3\sqrt{2}}{2} \end{pmatrix} = \begin{pmatrix} 0 \\ -3\sqrt{2} \end{pmatrix}$$

E por fim, $T_{13}(x, y) = \left(x \cdot \cos(-\pi) - y \cdot \text{sen}(-\pi), x \cdot \text{sen}(-\pi) + y \cdot \cos(-\pi)\right)$, é a transformação com $\theta = -\pi$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos(-\pi) & -\text{sen}(-\pi) \\ \text{sen}(-\pi) & \cos(-\pi) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} \cos(\pi) & +\text{sen}(\pi) \\ -\text{sen}(\pi) & \cos(\pi) \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 3 \end{pmatrix} = \begin{pmatrix} -3 \\ -3 \end{pmatrix}$$

Para verificar como funciona a rotação, a Figura 31 apresenta a rotação de um quadrado de lado uma unidade. Este quadrado foi rotacionado 7 vezes, através da variação do parâmetro θ de $[0, 2\pi]$, em intervalos de $\frac{\pi}{4}$.

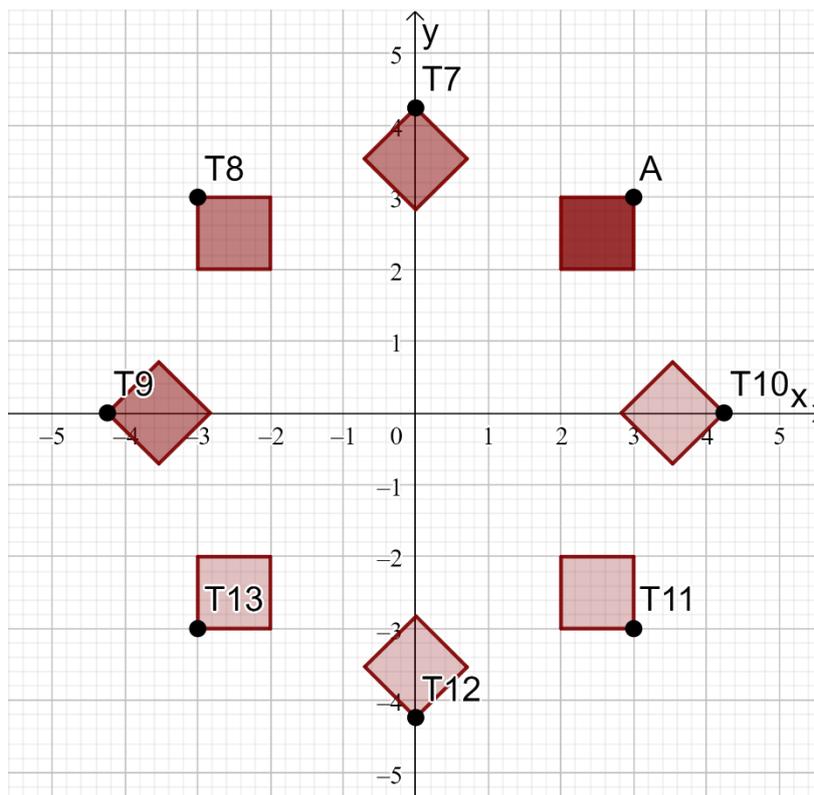


Figura 31: Figura elaborada no Geogebra para exemplificar as Transformações de 7 a 13.
 Fonte: Figura elaborada no Geogebra.

Observe que a rotação é realizada em torno da origem do sistema de coordenadas, mas não necessariamente com o objeto localizado na origem, quando isso acontece, o objeto também sofre outra transformação chamada de translação. Se o objeto estivesse localizado na origem, Schwartz (2016) destaca que ele não sofreria com o “efeito da translação” analisando a figura com base nos eixos do sistema de coordenada, como mostra a Figura 32.

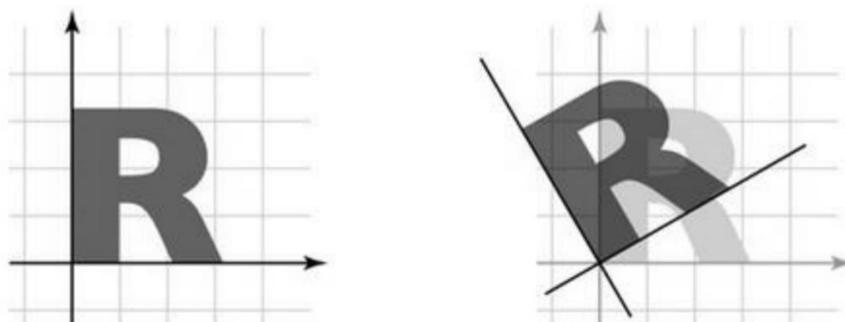


Figura 32: Figura da transformação de rotação de uma figura no centro do sistema de coordenadas.
 Fonte: Schwartz (2016) apud <http://slideplayer.com/slide/4875172/> (Acesso em 25/10/2016).

3.2.3 Reflexão

A reflexão é a transformação linear responsável por espelhar um vetor ou uma figura com relação a um dos eixos ou até mesmo aos dois eixos simultaneamente.

A transformação geométrica em questão possui parâmetro (-1) e, como o número 1 é o elemento neutro da multiplicação, essa aplicação não dilata, nem contrai o objeto, apenas o reflete, uma vez que produz total ou parcialmente, novas coordenadas com valores opostos aos valores das coordenadas originais. Vejamos alguns tipos de reflexões abaixo.

A transformação linear $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ é:

- i. Uma reflexão em torno do eixo x quando a transformação é dada por:

$$T(x, y) = (x, -y).$$

Assim a matriz dessa transformação é dada por:

$$T = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Dado um vetor inicial, podemos aplicar uma reflexão em torno do eixo x e obter as coordenadas finais do vetor transformado, usando a seguinte expressão matricial:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 \cdot x \\ (-1) \cdot y \end{pmatrix} = \begin{pmatrix} x \\ -y \end{pmatrix}$$

Considere o exemplo em que os vetores $A = (1,2)$ e $B = (2,2)$, sofrem uma reflexão, $T_{14}(x, y) = (x, -y)$, ao redor do eixo das abcissas.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 1 \\ -2 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ -2 \end{pmatrix}$$

- ii. Uma reflexão em torno do eixo y , se a transformação é $T(x, y) = (-x, y)$. Consequentemente, a matriz da transformação nesse caso, é dada por:

$$T = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Podemos estruturar essa transformação com base nas coordenadas iniciais do vetor, ou figura geométrica da seguinte forma:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} (-1) \cdot x \\ 1 \cdot y \end{pmatrix} = \begin{pmatrix} -x \\ y \end{pmatrix}$$

Analise o exemplo abaixo, em que a os vetores A e B sofrem a reflexões em torno do eixo y , ou seja, sua aplicação é da forma $T_{15}(x, y) = (-x, y)$

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} -1 \\ 2 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} -2 \\ 2 \end{pmatrix}$$

A Figura 33 a seguir mostra como se comportam as figuras quando sofrem reflexões ao redor dos eixos.

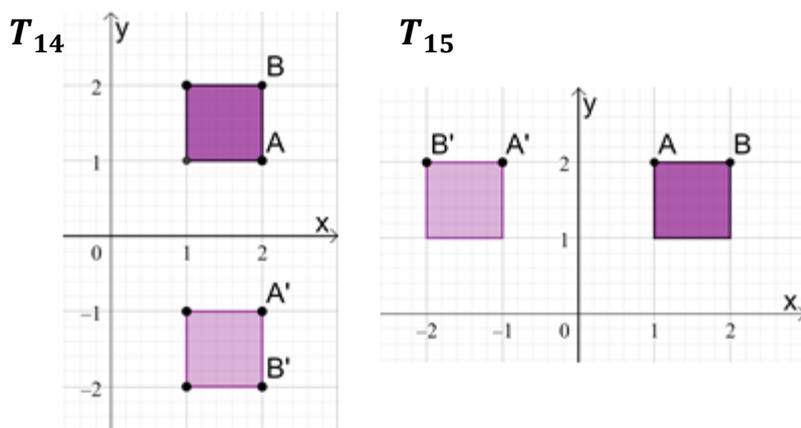


Figura 33: Figura elaborada no Geogebra para exemplificar reflexões ao redor dos eixos.
Fonte: Figura elaborada no Geogebra.

- iii. Caso a reflexão seja da forma $T(x, y) = (-x, -y)$, temos uma reflexão com base na origem do sistema de coordenadas. A matriz da transformação é:

$$T = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$$

É possível mapear a reflexão na origem por:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} (-1) \cdot x \\ (-1) \cdot y \end{pmatrix} = \begin{pmatrix} -x \\ -y \end{pmatrix}$$

Observe um exemplo desse mapeamento, em que vetores usados nas transformações 14 e 15 sofrem uma nova transformação em torno da origem.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} (-1) \cdot 1 \\ (-1) \cdot 2 \end{pmatrix} = \begin{pmatrix} -1 \\ -2 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} (-1) \cdot 2 \\ (-1) \cdot 2 \end{pmatrix} = \begin{pmatrix} -2 \\ -2 \end{pmatrix}$$

Os vetores transformados possuem extremidade nos pontos $A' = (-1, -2)$ e $B' = (-2, -2)$ como mostra a Figura 30.

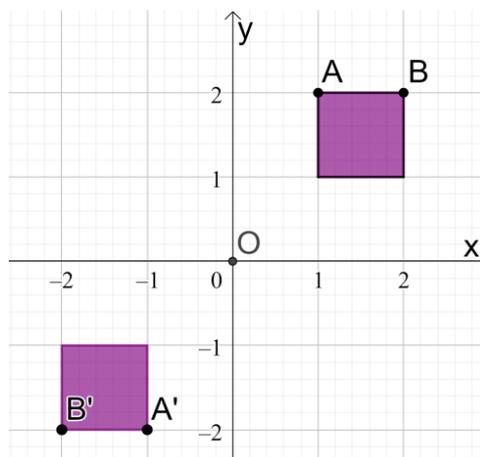


Figura 34: Figura Elaborada no Geogebra para exemplificar a reflexão em torno da origem.
Fonte: Figura Elaborada no Geogebra.

- iv. E por fim, para reflexões em torno da reta $y = x$, a transformação tem a forma $T(x, y) = (y, x)$, cuja matriz da transformação é dada por:

$$T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

É possível organizar a reflexão em torno da reta $y = x$ pela expressão:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \cdot x + 1 \cdot y \\ 1 \cdot x + 0 \cdot y \end{pmatrix} = \begin{pmatrix} y \\ x \end{pmatrix}$$

Observe na Figura 35 uma reflexão em torno da reta $y = x$, partindo dos mesmos vetores A e B .

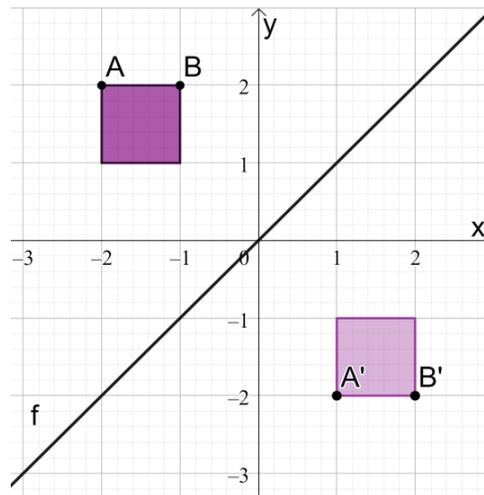


Figura 35: Figura elaborada no Geogebra como suporte para exemplificar uma reflexão.
 Fonte: Figura elaborada no Geogebra.

3.2.4 Cisalhamento

A transformação que desloca todos os pontos de uma direção fixa, cujo deslocamento é proporcional à sua distância a uma determinada paralela é chamada de cisalhamento.

Essa transformação geométrica causa a distorção das coordenadas de um objeto e pode ser realizada tanto na vertical quanto na horizontal.

A transformação linear $T: R^2 \rightarrow R^2$ é:

- i. Chamada de cisalhamento vertical quando $T(x, y) = (x, y + \alpha \cdot x)$ e em decorrência disso possui a seguinte matriz transformação:

$$T = \begin{bmatrix} 1 & 0 \\ \alpha & 1 \end{bmatrix}$$

A partir da seguinte expressão, podemos encontrar os valores iniciais transformados das coordenadas ou do vetor.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \alpha & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x + 0 \\ \alpha \cdot x + y \end{pmatrix} = \begin{pmatrix} x \\ \alpha x + y \end{pmatrix}$$

- ii. Chamada de cisalhamento horizontal quando é dada por $T(x, y) = (x + \alpha y, y)$ e em decorrência disso possui a seguinte matriz transformação:

$$T = \begin{bmatrix} 1 & \alpha \\ 0 & 1 \end{bmatrix}$$

Os pontos transformados podem ser encontrados em função dos pontos iniciais por meio da seguinte estrutura:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x + \alpha \cdot y \\ 0 + y \end{pmatrix} = \begin{pmatrix} x + \alpha \cdot y \\ y \end{pmatrix}$$

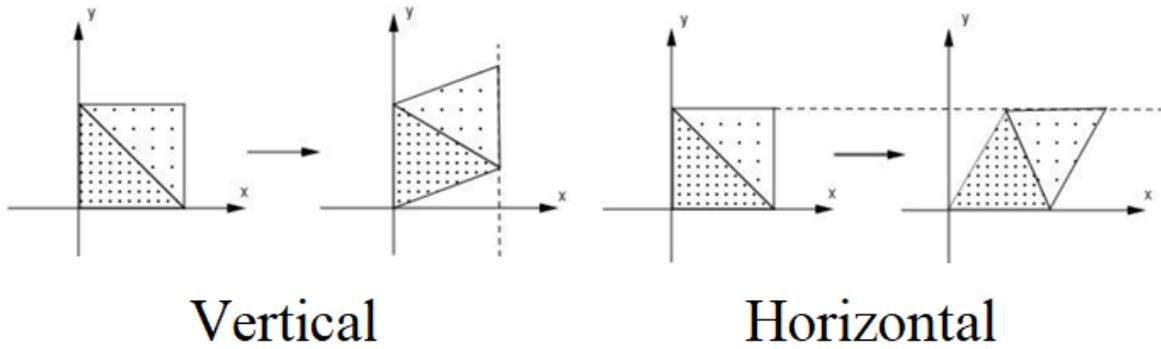


Figura 36: Figura para exemplificar os cisalhamentos.
Fonte: Schwartz (2016).

4. PROGRAMAÇÃO

A programação tem um papel fundamental na movimentação do Veículo Guiado Automaticamente, pois ela possibilita o funcionamento e o controle do Arduino, que por sua vez, por meio Linguagem C++ comanda todos os outros componentes da montagem.

A linguagem C++ foi desenvolvida em meados de 1980 por Bjarne Stroustrup e surgiu como uma versão aprimorada da Linguagem C, conservando a maioria de suas características, entre elas o fato de ser uma linguagem de nível intermediário à alto. De acordo com Trevelin (2012) uma linguagem de alto nível pode ser entendida como uma linguagem mais próxima da linguagem humana, ele ainda acrescenta:

[...] Fortran e Basic são consideradas linguagens de alto nível, pois seus comandos parecem-se com frases humanas (em inglês, claro): “Se $x = y$ então faça $x = x+1$ e imprima y ”. De maneira análoga, quanto mais próximo da linguagem da máquina for a linguagem de programação, mais baixo será seu nível: por exemplo, o Assembly é considerada uma linguagem de nível baixo, pois seus comandos são escritos em hexadecimal. (TREVELIN, 2012, p.3)

Isso é importante porque permite ao programador entender tudo que está fazendo em uma linguagem simples e mais próxima da usada em seu cotidiano. Neste capítulo estudaremos a programação utilizada no kit *Robot Car Starter*, incluindo as funções que permitem ao Arduino receber os sinais dos sensores e determinar como as rodas se movimentarão de modo que o veículo siga a linha preta.

4.1. Principais comandos

O comando *string*, de acordo com a página oficial do Arduino na internet, denomina-se uma cadeia de textos criada a partir de um vetor ou uma matriz (*array*) com apenas uma linha cujos dados são do tipo *char*.

Na tabela 2, note que nome representa cada tipo de dado usado na programação em Arduino.

Tabela 2: Tipo de dados e valores que podem ser atribuídos a eles.

Nome	Tipo de dado
<i>Boolean</i>	Valor verdadeiro ou falso
<i>Char</i>	Caracteres
<i>Byte</i>	Um byte, ou uma sequência
<i>Int</i>	Número inteiro de 16 bits com sinal
<i>unsigned int</i>	Número inteiro de 16 bits sem sinal
<i>Long</i>	Número inteiro de 16 bits com sinal
<i>unsigned long</i>	Número inteiro de 16 bits sem sinal
<i>Float</i>	Número real de precisão simples
<i>Double</i>	Número real de precisão dupla
<i>String</i>	Sequência de caracteres

Fonte: Adaptado de Backes (2012) e Trindade (2010).

Outro comando importante que aparece na programação é o *digitalRead*. Conforme Evans (2013), esse comando permite ler um valor por meio de um pino digital. Sendo que os valores possíveis são *high* ou *low*, o que respectivamente significa alto, cuja tensão é de 5V e baixo, quando tensão é 0V. Conforme observado na Tabela 3.

Tabela 3: Valor obtido pela leitura do pino e seu significado.

Valor	Significado
<i>High</i>	5V ou 3,3V
<i>Low</i>	0V

Elaborada no Excel e de autoria própria.

Além de fazer leituras com a função *digitalRead*, podemos também determinar como *high* ou *low* o valor de um pino digital configurado como *output* (saída) usando a função *digitalWrite*. Ou seja, quando o pino é designado como de saída, podemos variar a tensão para executar tarefas diferentes, como por exemplo, acender e apagar um LED.

Usando o comando *analogWrite*, ainda conseguimos escrever um valor analógico em um pino. No programa do veículo usado neste trabalho, essa função

serve para ativar e desativar os motores. E por fim, temos o comando *pinMode* que configura o pino para se comportar como entrada ou como saída, o que também pode ser observado na Tabela 4.

Tabela 4: Configuração dos pinos com a função "pinMode".

Configuração	Significado
<i>Input</i>	Entrada
<i>Output</i>	Saída

Elaborada no Excel e de autoria própria.

4.1.1. Desvio condicional

Um desvio condicional bastante usado na programação em Arduino é o comando *if*, que é descrito pela página oficial do Arduino na internet como uma “instrução que verifica uma condição e executa a instrução ou conjunto de instruções, se a condição for verdadeira”.

O fluxograma abaixo descreve o funcionamento dessa estrutura.

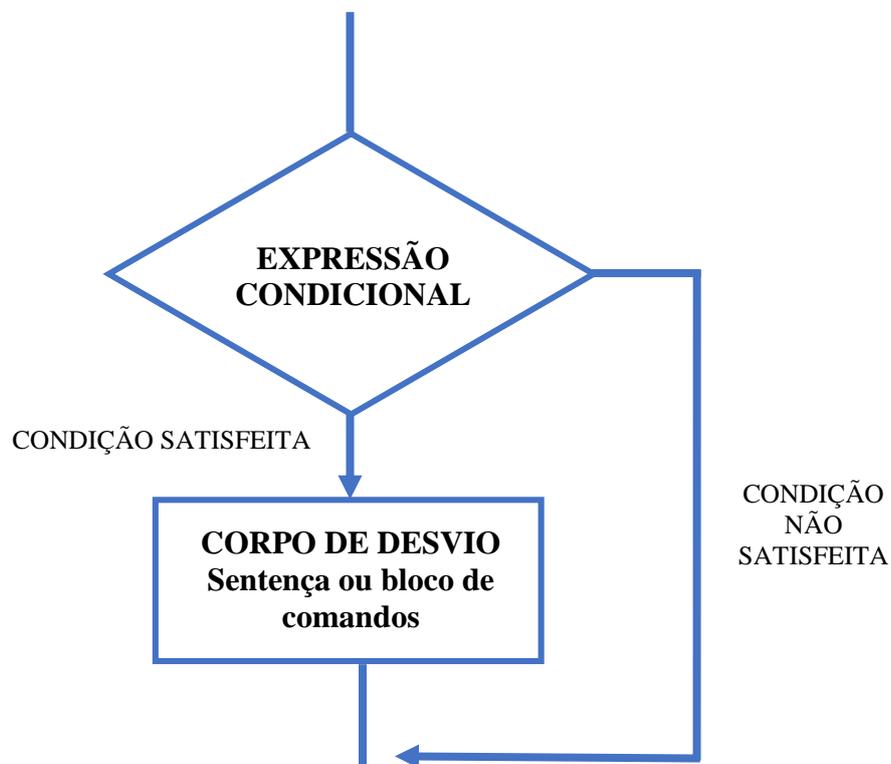


Figura 37: Fluxograma que descreve o funcionamento do IF.
Fonte: Elaboro no Word (2010) e de autoria própria.

Normalmente para descrever a expressão condicional do if são usados operadores lógicos. A tabela 5 traz alguns destes operadores e seus significados.

Tabela 5: Operadores lógicos na linguagem C/C++ e seus significados.

Operador	Significado
+	Soma
-	Subtração
/	Divisão
*	Multiplicação
^	Potência
<	Menor
<=	Menor ou igual
>	Maior
>=	Maior ou igual
==	Igual
!=	Diferente
&&	E
	Ou

Adaptado de Backes (2012) e Trindade (2010).

4.2. Fluxograma

O fluxograma abaixo descreve resumidamente o que o código de programação do AGV executa.

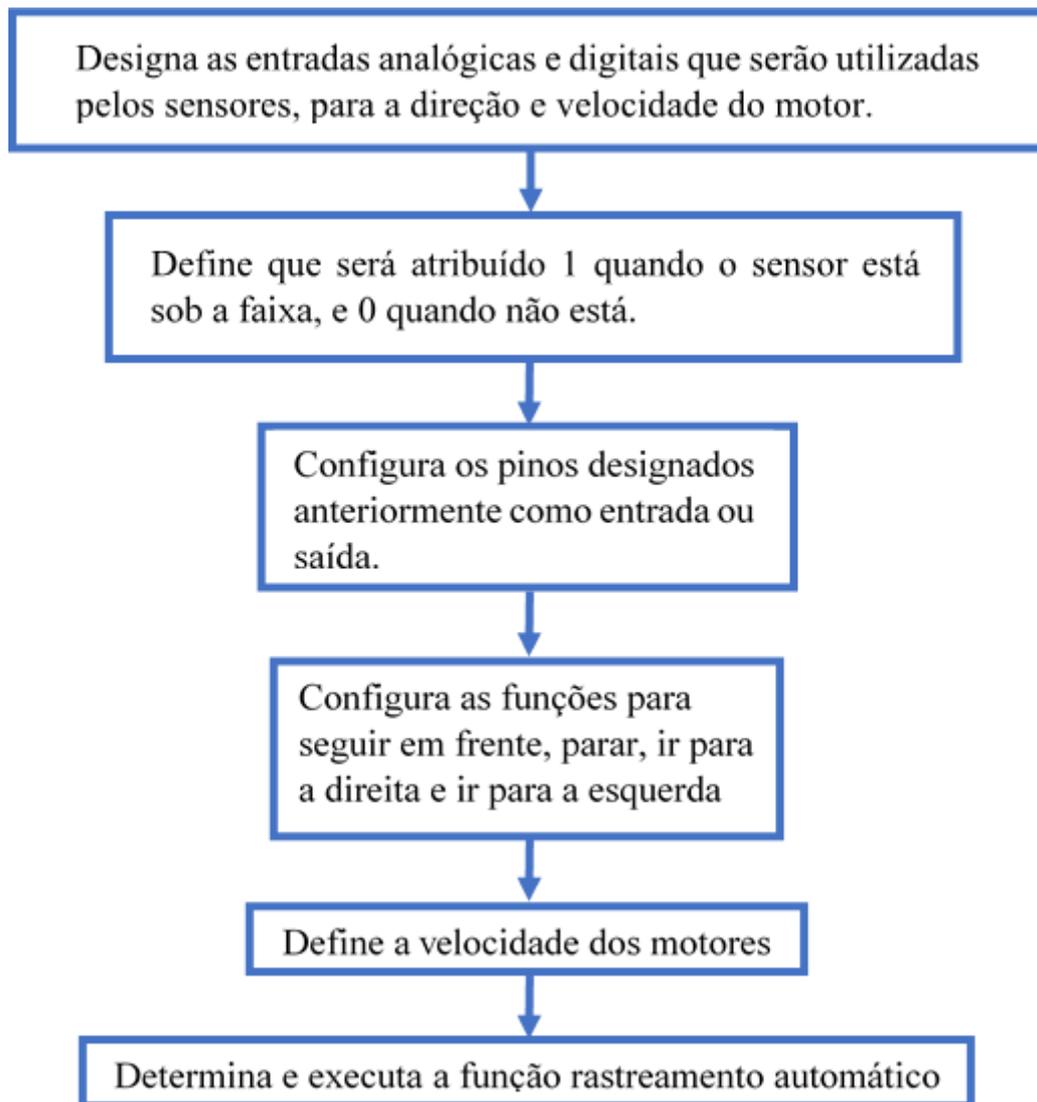


Figura 38: Fluxograma que mostra a ordem e o que a programação executa. Elaborado no Word (2010) e autoria própria.

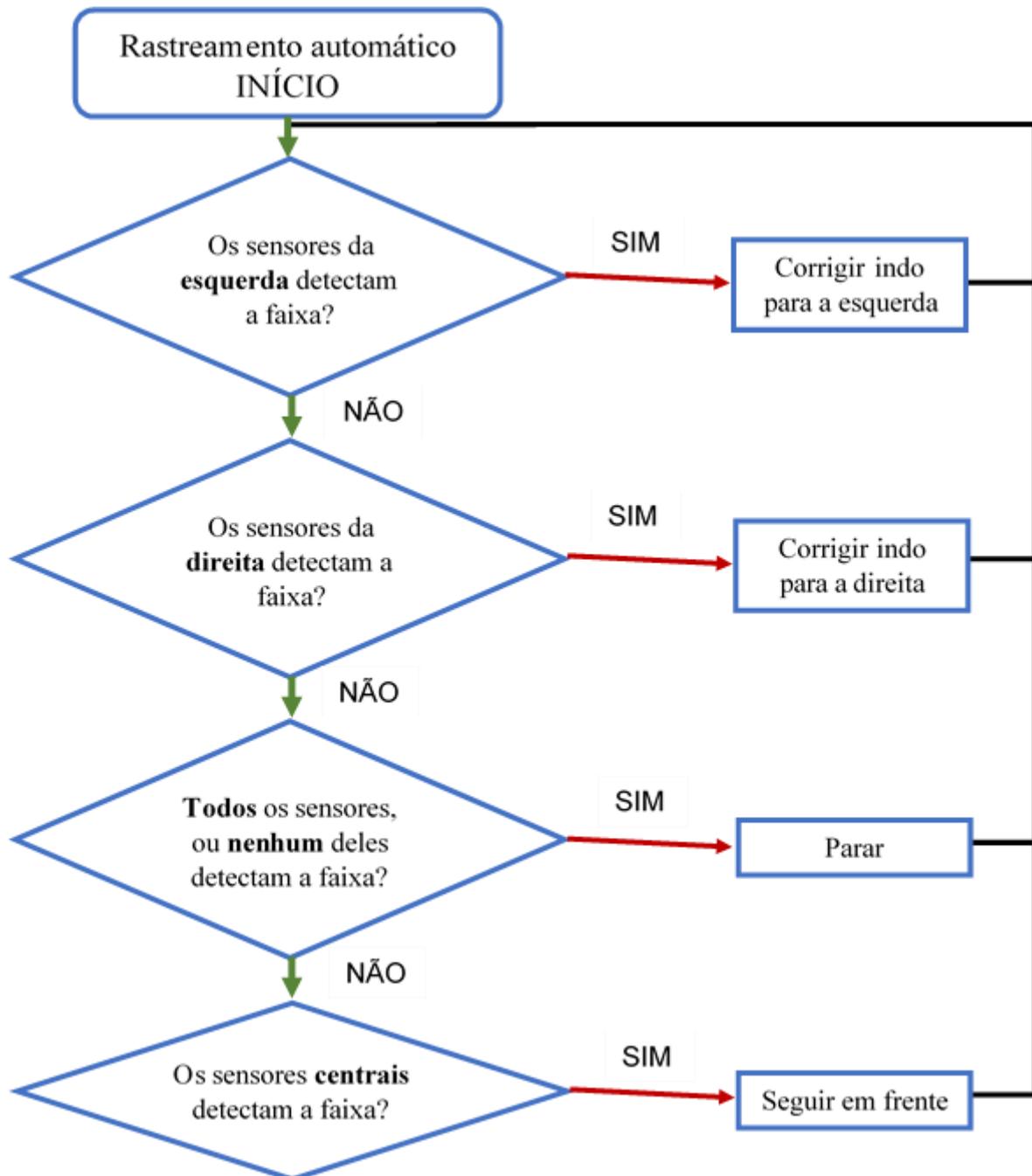


Figura 39: Fluxograma que mostra o que a função Rastreamento automático executa conforme as respostas dos sensores. Elaborada no Word (2010) e autoria própria.

5. MATRIZES NA DESCRIÇÃO DO TRAJETO PERCORRIDO POR UM AGV

Utilizando as Transformações Lineares explanadas no capítulo três e a programação descrita no capítulo quatro para controlar a movimentação do Veículo Guiado Automaticamente detalhado no segundo capítulo, analisaremos agora como as matrizes podem ser usadas na prática para descrever o trajeto percorrido por um veículo no plano cartesiano.

Um veículo automático é ideal para esta aplicação, pois garante a flexibilidade e a diversidade de trajetos, uma vez que, trocando a pista ou mudando a disposição da faixa temos novos caminhos e novas operações matriciais podem ser trabalhadas.

5.1. Dilatação

Antes de descrever a movimentação do veículo por um trajeto por meio da dilatação, considere que nas transformações geométricas exploradas no terceiro capítulo e nas transformações a seguir usamos um recurso muito utilizado para facilitar os cálculos. Esse recurso é descrito por Steinbruch (1987) que deixa claro que qualquer vetor \overrightarrow{AB} , ou seja, qualquer vetor de origem A e extremidade B , analisado em um sistema de coordenadas, sempre tem um representante com mesmo comprimento, direção e sentido, cuja ponto inicial é a origem do sistema como mostra a Figura 40, está propriedade é conhecida como equipolência entre vetores.

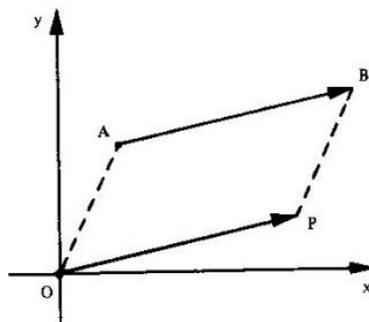


Figura 40: Representante do Vetor \overrightarrow{AB} com extremidade na origem do plano cartesiano. Steinbruch e Winterle (1987).

Dessa forma, cada vetor no plano é determinado apenas pelo ponto extremo do segmento e o ponto P caracteriza o vetor $v = \overrightarrow{OP}$, que pode ser descrito como $v = (x, y)$. Por esta razão, a dilatação abaixo é restrita a vetores diferentes do vetor nulo, ou seja, o início da faixa não deve estar na origem do sistema cartesiano. Isso porque este vetor teria comprimento 0 e a dilatação sobre a componente que é nula não ocorrerá.

Além disso, como já citado no capítulo 3, neste trabalho, ao considerar alguns pontos do plano cartesiano, não estamos considerando apenas as coordenadas do ponto em questão, estamos analisando o vetor, ou seja, o segmento orientado com comprimento, direção e sentido, que se inicia na origem do plano cartesiano e possui extremidade no ponto estudado.

Tendo em vista o que foi discutido anteriormente, é possível observar a Transformação Linear denominada por dilatação ao posicionar uma faixa reta no plano cartesiano e colocar o veículo para percorrer essa faixa, uma vez que os pontos da faixa representam a extremidade de vetores que têm início na origem do sistema de coordenadas. Com base no que já foi explorado sobre dilatação, também podemos afirmar que se a faixa é vertical a transformação possui um parâmetro $\beta > 0$ e caso contrário, ou seja, se a faixa é horizontal a transformação terá um parâmetro $\alpha > 0$.

Analisemos a seguir a transformação que descreve a movimentação do veículo sob uma faixa horizontal, posicionada no sistema de coordenadas, que se inicia no ponto $A = (2,0)$ e sofre uma dilatação cujo parâmetro $\alpha = 10$.

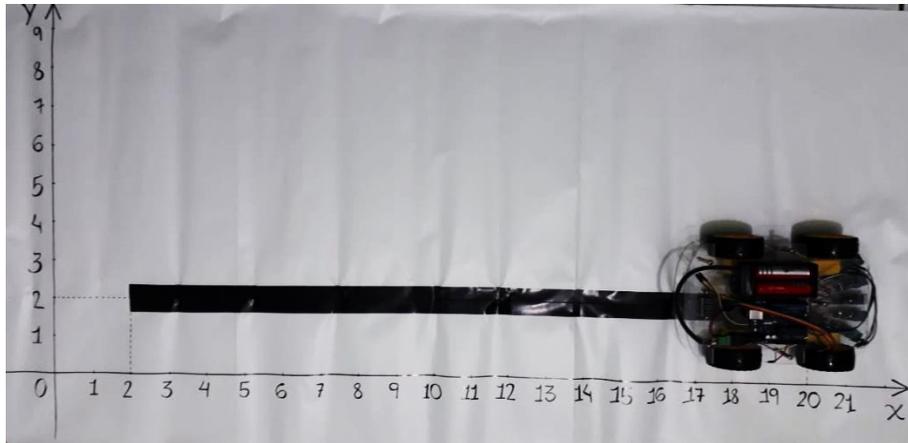
$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} 10 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 20 \\ 0 \end{pmatrix}$$

Note que o seguimento que dá origem a transformação acima termina em $C = (20,0)$.

Analogamente, observe agora outra dilatação com o mesmo parâmetro, mas sob uma faixa iniciada em $B = (2,2)$.

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} 10 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 20 \\ 2 \end{pmatrix}$$

Agora note que o seguimento acaba em $D = (20,2)$. Conforme a Figura 41 ilustra.



*Figura 41: Movimentação por uma faixa na horizontal.
Fotografia do veículo na pista confeccionada.*

Atente-se agora para a dilatação de parâmetro $\beta = 5$ sob um vetor cuja extremidade coincide com o início de uma a faixa uma faixa vertical em $A = (0,2)$ e outra dilatação com o mesmo parâmetro, mas sob o vetor $B = (2,2)$.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 10 \end{pmatrix}$$

$$\begin{pmatrix} x'_B \\ y'_B \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 10 \end{pmatrix}$$

Observe na Figura 41 as movimentações do veículo descritas por meio das dilatações acima. Com a transformação linear que chamamos de contração podemos fazer a mesma aplicação da dilatação, uma vez que a única diferença entre elas é o parâmetro, entretanto ambas alteram a escala do vetor posição.



Figura 42: Movimentação do veículo sob uma faixa vertical.
Fotografia do veículo na pista confeccionada.

5.2. Rotação

A rotação é a Transformação Linear que pode ser usada para descrever os trechos curvos de um trajeto percorrido por um robô.

Na figura 43 podemos observar uma circunferência cujo centro está posicionado na origem do plano cartesiano, sendo que o veículo percorrerá o arco limitado pelo eixo x no ponto $A = (4,0)$ e o ponto A' no sentido anti-horário com o parâmetro é $\theta = \frac{\pi}{2}$.

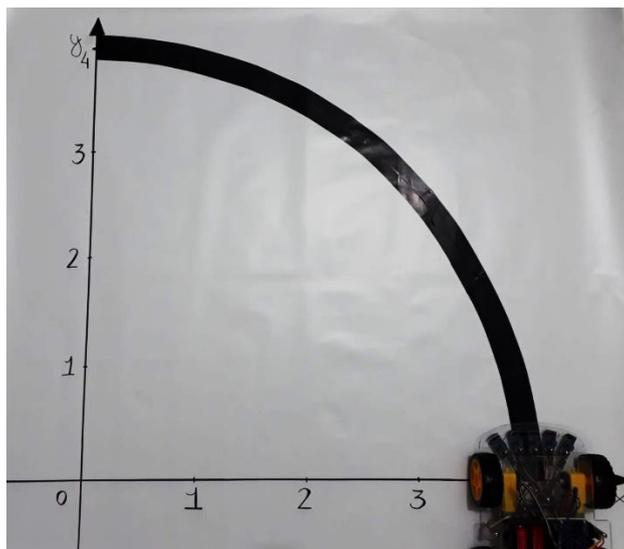
$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{2}\right) & -\text{sen}\left(\frac{\pi}{2}\right) \\ \text{sen}\left(\frac{\pi}{2}\right) & \cos\left(\frac{\pi}{2}\right) \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \end{pmatrix}$$

Logo, o veículo irá percorrer um quarto da circunferência no sentido anti-horário, partindo do ponto $A = (4,0)$ e chegando em $B = (0,4)$.

Veja agora outra rotação nesta circunferência com mesmo ponto de partida e parâmetro, mas no sentido horário.

$$\begin{pmatrix} x'_A \\ y'_A \end{pmatrix} = \begin{pmatrix} \cos\left(-\frac{\pi}{2}\right) & -\text{sen}\left(-\frac{\pi}{2}\right) \\ \text{sen}\left(-\frac{\pi}{2}\right) & \cos\left(-\frac{\pi}{2}\right) \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 & +1 \\ -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ -4 \end{pmatrix}$$

Se percorrer um arco de $\theta = \frac{\pi}{2}$ mas no sentido horário o veículo chegará no ponto $A' = (0, -4)$.



*Figura 43: Movimentação do veículo na circunferência.
Fotografia do veículo na pista confeccionada.*

5.3. Reflexão

Ainda podemos usar as informações de saída e chegada do veículo para encontrar a matriz transformação. Em exemplo disso, é aplicado a seguir com a movimentação do veículo ao longo da diagonal que coincide com a bissetriz do primeiro e do terceiro quadrante do plano cartesiano, usando a transformação conhecida por reflexão em torno da reta $y = x$.

Com base nisso, observe como é descrito o trajeto do veículo que partiu do ponto $A = (-5,6)$ e chegou ao ponto $A' = (6,-5)$.

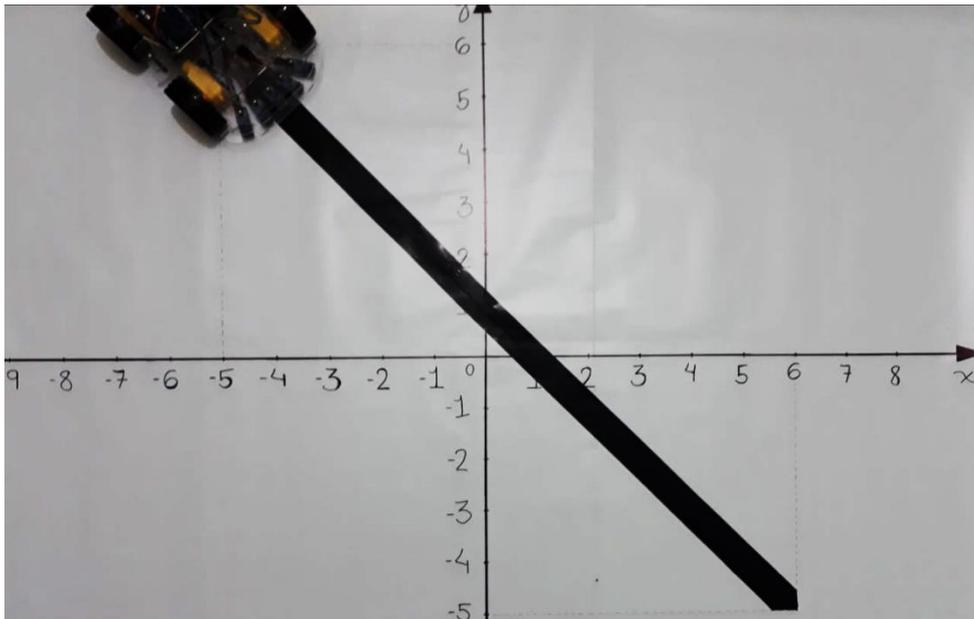


Figura 44: A figura mostra a movimentação do veículo que será usada para verificar a matriz que descreve este movimento.

Fonte: Fotografia do veículo na pista confeccionada.

Usaremos uma matriz genérica com elementos a, b, c e d para verificar qual é a matriz que descreve a movimentação realizada pelo veículo na Figura 44.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \Rightarrow \begin{pmatrix} 6 \\ -5 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} -5 \\ 6 \end{pmatrix} \Leftrightarrow \begin{pmatrix} 6 \\ -5 \end{pmatrix} = \begin{pmatrix} -5a + 6b \\ -5c + 6d \end{pmatrix}$$

A igualdade de matrizes acima só é verdadeira quando a e d são iguais a zero e c e b iguais a um. Logo, a matriz da transformação de reflexão em torno da reta $y = x$ é dada por:

$$T = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Podemos repetir esse processo para encontrar a matriz da transformação conhecida como reflexão em torno da origem usando as informações de um veículo que parte de um ponto $A = (8, 9)$ e após andar de acordo a com a reflexão em torno da origem chega no ponto $A' = (-8, -9)$.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \Rightarrow \begin{pmatrix} -8 \\ -9 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} 8 \\ 9 \end{pmatrix} \Leftrightarrow \begin{pmatrix} -8 \\ -9 \end{pmatrix} = \begin{pmatrix} 8a + 9b \\ 8c + 9d \end{pmatrix}$$

Usando igualdade de matrizes concluímos a e d são iguais a um negativo e c e b iguais à zero. Consequentemente, a matriz da transformação de reflexão em torno da origem seguinte:

$$T = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$$

5.4. Transformações compostas

Nos tópicos anteriores conseguimos concluir o ponto de chegada de um veículo que caminha no plano cartesiano segundo uma transformação conhecida, ou de posse dos pontos de partida e chegada de um veículo determinar a matriz que descreve o trajeto que ele percorreu.

No entanto, as propriedades que foram usadas até o momento para trabalhar as transformações ainda podem ser aplicadas a mais de uma transformação de uma só vez usando a transformação composta. Para aplicar a transformação linear composta precisamos considerar um teorema importante.

Teorema: *Considere U, V e W espaços vetoriais, com as respectivas bases B, C e D . Sejam as transformações lineares $F: U \rightarrow V$ e $G: V \rightarrow W$. A matriz da transformação linear composta $G \circ F: U \rightarrow W$ é dado por:*

$$(G \circ F)_{B,D} = (G)_{C,D}(F)_{B,C}$$

O teorema acima afirma que a Matriz da transformação linear composta é dada pelo produto das matrizes de cada transformação.

Com base nisso, vamos determinar a movimentação de um veículo que parte de $A = (-8, 2)$ e percorre uma faixa reta até $B = (-2, 2)$. A partir deste ponto passa pela reta $y = -x$ até $C = (2, -2)$ onde começa a percorrer outra reta e chega no ponto $D = (8, -2)$.

Primeiramente vamos definir a matriz da transformação que leva o veículo do ponto A até o ponto B . Como $x_A \neq x_B$ e $y_A = y_B$, temos para esta primeira transformação que a faixa é horizontal e é percorrida da esquerda para a direita já que o valor de x aumenta. Então, sabendo o que a extremidade do vetor acompanha a movimentação do veículo, podemos concluir que a transformação em questão é

uma contração uma vez que a extremidade do vetor se aproxima da origem depois que ele é transformado.

Tendo em vista a contração, basta determinar os valores de α e β para obter a matriz desta primeira transformação.

Esta transformação corresponde a uma contração na direção x e que o y se mantém constante, de T_1 temos: $\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha x \\ \beta y \end{pmatrix}$, em que $0 < \alpha < 1$ e

$$\beta = 1.$$

Sendo $A = (-8, 2) = (x, y)$ e $B = (-2, 2) = (x', y')$, temos em T_1 :

$$\begin{pmatrix} -2 \\ 2 \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} -8 \\ 2 \end{pmatrix} = \begin{pmatrix} -8\alpha \\ 2 \end{pmatrix} \Leftrightarrow \begin{pmatrix} -2 \\ 2 \end{pmatrix} = \begin{pmatrix} -8\alpha \\ 2\beta \end{pmatrix}$$

Assim temos que $-8\alpha = -2 \Leftrightarrow \alpha = \frac{2}{8} = \frac{1}{4}$. E dessa forma, a matriz de T_1 é:

$$T_1 = \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{bmatrix}$$

Agora, voltemos a atenção para o trecho do percurso que sai de B e chega em C pela reta $y = -x$. Para esta transformação, temos de modo geral que:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \Leftrightarrow \begin{pmatrix} -x \\ -y \end{pmatrix} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix}$$

E pela igualdade de matrizes obtemos b e c iguais a zero e a e d iguais a um negativo. Dessa forma a matriz geral desta transformação é conhecida como a matriz da transformação de reflexão em torno da origem:

$$T_2 = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$$

E por fim, vamos determinar a matriz da transformação referente ao trecho que descreve a parte do trajeto que parte de C e segue até D .

Novamente temos $x_A \neq x_B$ e $y_A = y_B$, então essa parte do percurso também é horizontal. Mas como o x aumenta e se afasta da origem à medida que o vetor é transformado, desta vez temos uma dilatação.

Então a transformação que o veículo sofre é uma dilatação e basta determinar o valor de α para obter a matriz desta primeira transformação.

Como T_3 é uma contração na direção x e que o y se mantém, temos:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha x \\ \beta y \end{pmatrix}, \text{ em que } \alpha < 1 \text{ e } \beta = 1.$$

Utilizando os pontos C e D temos:

$$\begin{pmatrix} 8 \\ -2 \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ -2 \end{pmatrix} = \begin{pmatrix} 2\alpha \\ -2 \end{pmatrix}$$

Assim temos que $2\alpha = 8 \Leftrightarrow \alpha = \frac{8}{2} = 4$. E dessa forma, a matriz de T_3 é:

$$T_3 = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$$

De acordo com o teorema podemos compor mais de uma transformação linear ao multiplicar todas as transformações envolvidas. No entanto, lembrando que a operação de multiplicação de matrizes não é comutativa, temos que realizar as multiplicações na ordem com que elas são realizadas e, como o último fator é a matriz com as coordenadas iniciais, os outros fatores desta composição, tem que ser organizados em ordem, começando com a matriz da última transformação, até a matriz da primeira transformação que será realizada.

Com base nisso, para descrever todo o trajeto percorrido pelo veículo, temos que compor as transformações T_1, T_2 e T_3 . Consequentemente, a transformação que descreve todo o percurso do veículo no plano cartesiano é:

$$T = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 4 & 1 \end{bmatrix}$$

E aplicando as coordenadas iniciais $A = (-8, 2)$ do veículo conseguimos obter as coordenadas de parada $D = (2, -8)$ dele por meio dessa expressão.

$$T = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 4 & 1 \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1/4 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -8 \\ 2 \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} -4 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1/4 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -8 \\ 2 \end{pmatrix} \Leftrightarrow \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} -8 \\ 2 \end{pmatrix} \Leftrightarrow \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 8 \\ -2 \end{pmatrix}$$

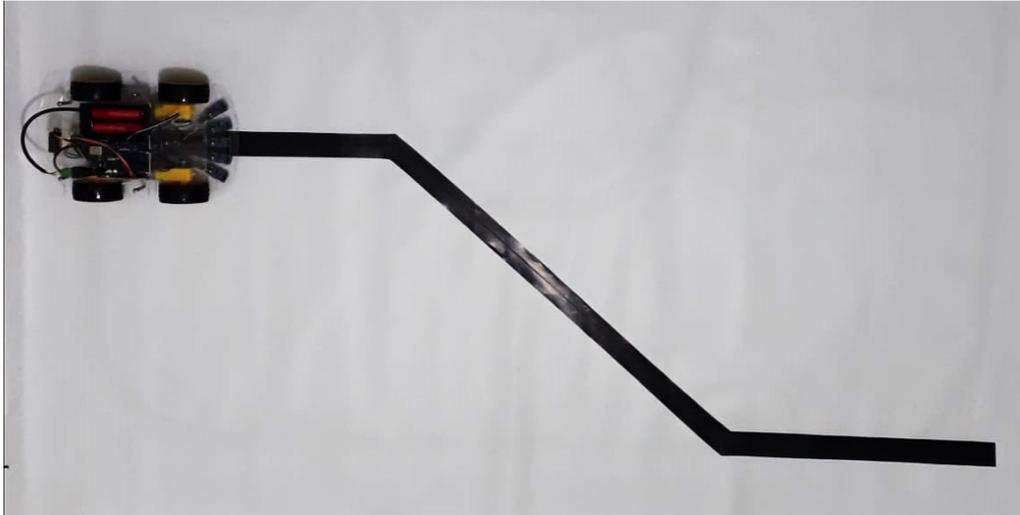


Figura 45: Parte da movimentação descrita por transformações compostas. Fotografia do veículo na pista confeccionada.

As transformações geométricas podem ser usadas juntamente com a movimentação de um veículo disposto no plano cartesiano para explorar as operações e as principais propriedades das matrizes.

No entanto, apesar do objetivo deste trabalho ser mostrar essa relação usando um veículo automático, também podemos usar um veículo robótico programado previamente para realizar um trajeto predefinido e explorar as transformações envolvidas nesse trajeto. Um exemplo disso é programar o veículo para percorrer dois metros em linha reta, fazer uma rotação em torno de seu próprio eixo, cujo parâmetro é $\frac{\pi}{2}$, percorrer novamente dois metros e fazer novamente uma rotação com o mesmo parâmetro e repetir isso mais duas vezes. Neste exemplo temos um quadrado com lado igual a dois metros e podemos explorar as reflexões em torno do eixo x e y com ele.

5.4. Programação e algoritmo

Além de explorar a matemática das operações com matrizes, ainda podemos, com um Veículo Guiado Automaticamente, explorar a programação envolvida. Porque o aluno, ao entender como a programação funciona e como ela possibilita a movimentação do veículo, desenvolve um pensamento computacional, o que a Sociedade Brasileira de Computação (SBC) *apud* Garlet (2016), define como um

processo cognitivo desenvolvido pelos humanos que os capacita a resolver problemas. Vale lembrar que não se trata de fazer do estudante um programador, mas fazê-lo capaz de entender a lógica de programação e a sintaxe da linguagem.

A programação também ajuda o aluno a compreender melhor a finalidade e o funcionamento de algoritmos. Para Donald Knuth (1968) apud Setti (2009), um algoritmo é um conjunto de regras que introduz uma saída específica para um conjunto de entradas, ele realiza o processamento dessas entradas e após um número finito de passos, chega a um resultado. Partindo desse ponto de vista para algoritmos percebemos que a matemática é repleta de algoritmos, e a programação ajuda o estudante a refletir e entender melhor o funcionamento de algoritmos como o algoritmo de Euclides e os algoritmos usados em operações básicas como multiplicação e divisão.

CONCLUSÃO

Este trabalho articulou conceitos e procedimentos abordados em álgebra linear e robótica, utilizando as transformações lineares, linguagem de programação e um veículo guiado automaticamente. Ele mostrou como a álgebra linear por meio de transformações lineares como reflexão, rotação, dilatação e suas compostas estão presentes na movimentação de um Veículo Guiado Automaticamente no plano cartesiano, evidenciando que a programação de um veículo com essas particularidades pode contribuir para a autonomia dos estudantes para solucionar problemas, contribuindo com o desenvolvimento do raciocínio lógico e do pensamento crítico, características essas, essenciais para o desenvolvimento do estudante.

Ao fim deste trabalho atingimos o objetivo inicial de descrever a movimentação observada no trajeto percorrido por um veículo guiado automaticamente por meio de operações com matrizes. E isso é importante porque tendo em vista o momento histórico em que vivemos é essencial que o professor utilize recursos tecnológicos e didáticos compatíveis com o momento para fornecer a formação completa que o estudante precisa, além disso, o professor deve apresentar a Matemática de forma desafiadora e lúdica. E para isto a robótica educacional pode ser um excelente ferramental porque por meio de atividades que associam elementos da robótica como programação, elétrica, eletrônica e mecânica, explora conteúdos matemáticos, como por exemplo, as transformações geométricas analisadas no decorrer neste trabalho.

Como uma futura evolução para este trabalho, a sugestão é que se desenvolva uma proposta didática coerente com o que foi explanado até o momento sobre a utilização da robótica educacional na sala de aula, como por exemplo, uma sequência didática para explorar a multiplicação e outras operações com matrizes por meio da movimentação no plano cartesiano para a educação básica. Atividades que permitam que o aluno tenha contato tanto com a parte robótica quanto com o conhecimento matemático e verifique por si só a influência de cada transformação geométrica em um objeto posicionado no plano cartesiano. Para isto, assim como aponta o trabalho, o Arduino é uma excelente ferramenta, uma vez que permite a

construção de projetos robóticos elaborados, é simples e de fácil manuseio e é financeiramente acessível a maioria dos ambientes escolares.

Anexo A - Programa comentado do Veículo Guiado Automaticamente

```
/*Os códigos de programa a seguir visam declarar o componente denominado pelo fabricante como L298N, que se refere a uma versão aprimorada do módulo Ponte H. Também define que finalidade terá cada pino desse módulo. */
```

```
#define dir1PinL 2 //Define a entrada digital D2 do Arduino como a direção do primeiro motor da esquerda
```

```
#define dir2PinL 4 //Define a entrada digital D4 do Arduino como a direção do segundo motor da esquerda
```

```
#define speedPinL 6 //Define o pino D6 como PWM para controlar a velocidade do motor esquerdo
```

```
#define dir1PinR 7 //Define a entrada digital D7 como a direção do primeiro motor da direita
```

```
#define dir2PinR 8 //Define a entrada digital D8 como a direção do segundo motor da direita
```

```
#define speedPinR 5 //Define o pino D5 como PWM para controlar a velocidade do motor direito
```

```
#define LFSensor_0 3 //Define a entrada digital D3 para o sensor 1 da esquerda
```

```
#define LFSensor_1 A1 //Define a entrada analógica A1 para o sensor 2 da esquerda
```

```
#define LFSensor_2 A2 //Define a entrada analógica A2 para o sensor do centro
```

```
#define LFSensor_3 A3 //Define a entrada analógica A3 para o sensor 2 a direita
```

```
#define LFSensor_4 10 //Define a entrada analógica A1 para o sensor 1 da esquerda
```

```
#define SPEED 150 //Define a velocidade no motor em 150
```

```
char sensor[5]; //Cria um vetor chamado sensor, com capacidade de armazenamento de 5 caracteres
```

```
/*Estabelece que o sensor lê 1 quando "enxerga" o preto e lê 0 quando "enxerga" o branco, define a sequencia de leitura dos sensores */
```

```

String read_sensor_values()
{
    int sensorvalue=32; //Criar e atribuir o valor 32 a uma variável inteira
    sensor[0]= digitalRead(LFSensor_0); //Lê o valor do pino LFSensor_0
    sensor[1]=digitalRead(LFSensor_1); //Lê o valor do pino LFSensor_1
    sensor[2]=digitalRead(LFSensor_2); //Lê o valor do pino LFSensor_2
    sensor[3]=digitalRead(LFSensor_3); //Lê o valor do pino LFSensor_3
    sensor[4]= digitalRead (LFSensor_4); //Lê o valor do pino LFSensor_4

    /*Atribui pesos aos valores determinados pela leitura dos sensores*/
    sensorvalue+= sensor[0]*16+sensor[1]*8+sensor[2]*4+sensor[3]*2+sensor[4];
    String senstr= String(sensorvalue,BIN); //Estabelece uma variável do tipo
    string
    return senstr.substring(1,6); //Termina a função e retorna um valor
}
void go_Advance(void) //Determina a função "Avance", para o veículo seguir a
frente
{
    digitalWrite(dir1PinL,LOW); //Aciona o valor LOW para o pino digital
designado para dir1PinL
    digitalWrite(dir2PinL,LOW); //Aciona o valor LOW para o segundo pino digital
da esquerda
    digitalWrite(dir1PinR,LOW); //Aciona o valor LOW para o primeiro pino digital
da direita
    digitalWrite(dir2PinR,LOW); //Aciona o valor LOW para o segundo pino digital
da direita
}

```

```
void go_Left(void) //Determina a função "Esquerda", para o veículo ir para a esquerda
```

```
{  
  
    digitalWrite(dir1PinL, HIGH); //Aciona o valor HIGH para o primeiro pino digital da esquerda  
  
    digitalWrite(dir2PinL,LOW); //Aciona o valor LOW para o segundo pino digital da esquerda  
  
    digitalWrite(dir1PinR,LOW); //Aciona o valor LOW para o primeiro pino digital da direita  
  
    digitalWrite(dir2PinR,HIGH); //Aciona o valor HIGH para o segundo pino digital da direita  
  
}
```

```
void go_Right(void) //Determina a função "Direita", para o veículo ir para a direita
```

```
{  
  
    digitalWrite(dir1PinL, LOW); //Aciona o valor LOW para o primeiro pino digital da esquerda  
    digitalWrite(dir2PinL,HIGH); //Aciona o valor HIGH para o segundo pino digital da esquerda  
  
    digitalWrite(dir1PinR,HIGH); //Aciona o valor HIGH para o primeiro pino digital da direita  
  
    digitalWrite(dir2PinR,LOW); //Aciona o valor LOW para o segundo pino digital da direita  
  
}
```

```
void go_Back(void) //Determina a função "Voltar", para o veículo andar na ré
```

```
{  
    digitalWrite(dir1PinL, LOW); //Aciona o valor LOW para o primeiro pino digital da esquerda  
  
    digitalWrite(dir2PinL,HIGH); //Aciona o valor HIGH para o segundo pino digital da esquerda  
  
}
```

```

digitalWrite(dir1PinR,LOW); //Aciona o valor LOW para o primeiro pino digital
da direita

digitalWrite(dir2PinR,HIGH); //Aciona o valor HIGH para o segundo pino digital
da direita

}

void stop_Stop() //Determina a função "Pare", para o veículo parar

{

    digitalWrite(dir1PinL, LOW); //Aciona o valor LOW para o primeiro pino digital
da esquerda

    digitalWrite(dir2PinL,LOW); //Aciona o valor LOW para o segundo pino digital
da esquerda

    digitalWrite(dir1PinR,LOW); //Aciona o valor LOW para o primeiro pino digital
da direita

    digitalWrite(dir2PinR,LOW); //Aciona o valor LOW para o segundo pino digital
da direita

}

void set_Motorspeed(int speed_L,int speed_R) //Definir velocidade do motor

{

    analogWrite(speedPinL,speed_L); //Determina um valor analogico para a
velocidade dos motores da esquerda

    analogWrite(speedPinR,speed_R); //Determina um valor analogico para a
velocidade dos motores da direita

}

void setup()

{

    /*Configurar os pinos PWM como saída*/

    pinMode(speedPinL,OUTPUT);

```

```

pinMode(speedPinR,OUTPUT);

/*Configurar os pinos como saída*/

pinMode(dir1PinL,OUTPUT);

pinMode(dir2PinL,OUTPUT);

pinMode(dir1PinR,OUTPUT);

pinMode(dir2PinR,OUTPUT);

/*Configurar os sensores como entrada*/

pinMode(LFSensor_0,INPUT);

pinMode(LFSensor_1,INPUT);

pinMode(LFSensor_2,INPUT);

pinMode(LFSensor_3,INPUT);

pinMode(LFSensor_4,INPUT);

Serial.begin(9600); //Determina a taxa de dados da porta serial: 9600 bps
}

/*Determina a função rastreamento automático*/

void auto_tracking() //Determina a função rastreamento automático
{

String sensorval= read_sensor_values();

/*Função que determina que se os sensores da esquerda lêem 1, ou seja,
estão sob a faixa, o veículo deve ir mais para a esquerda*/

if (sensorval=="11100" or sensorval=="11000" or sensorval=="10000" or
sensorval=="01000")
{

```

```

    go_Left();
    set_Motorspeed(0,SPEED);
}

/*Função que determina que se o veículo está a direita da faixa, deve ir mais
para a direita*/
if (sensorval=="00111" or sensorval=="00011" or sensorval=="00001" or
sensorval=="00010" )
{
    go_Right();
    set_Motorspeed(SPEED,0);
}

/*Função que determina que se o veiculo não está sob a faixa ou está sob
uma faixa que é captada pelos 5 sensores, deve parar */
if (sensorval=="00000" or sensorval=="11111" or sensorval=="01111" or
sensorval=="11110")
{
    stop_Stop();
    set_Motorspeed(0,0);
}

/*Função que determina que se a faixa está no meio do veículo, este deve
confiar em frente, sem corrigir para a esquerda nem para a direita*/
if (sensorval=="00100" or sensorval=="01110" or sensorval=="00110" or
sensorval=="01100")
{

```

```
        go_Advance();
        set_Motorspeed(SPEED,0);
        set_Motorspeed(0,SPEED);
    }
}
/*Garantir que a função auto_tracking, de rastreamento automático, seja executada
consecutivamente*/
void loop()
{
    auto_tracking();
}
/*Fim do programa*/
```


REFERÊNCIAS

- ALVES, R. M.; SILVA, A. L. C.; PINTO, M. C.; SAMPAIO, F. F.; ELIA, M. F. **Uso do hardware livre Arduino em ambientes de ensino-aprendizagem.** In: CONGRESSO BRASILEIRO DE INFORMÁTICA DA EDUCAÇÃO, 2012, Rio de Janeiro; JAIE - JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA E EDUCAÇÃO, 2012, Rio de Janeiro. Anais... Rio de Janeiro: CEIE, 2012.p. 162/187
- ANTON, H.; RORRES, C. **Álgebra linear com aplicações.** Ed. Bookman, 10. ed., Porto Alegre, 2012.
- BACKES, A. **Linguagem C: Completa e Descomplicada.** [S.l.]: Elsevier Editora Ltda., 2012. ISBN 9788535269642.
- BOLDRINI, J. L.; COSTA, S. I. R.; FIGUEIREDO, V. L.; WETZLER, H. G. **Álgebra linear.** Ed. Harbra Ltda, 3. ed., São Paulo, 1980.
- CALLIOLI, C. A.; COSTA, R. C. F.; DOMINGUES, H. H. **Álgebra Linear e Aplicações.** Atual Editora, São Paulo-SP, 1992.
- CASTELLS, M. **A Sociedade em Rede: do Conhecimento à Política.** In: CARDOSO, G.; CASTELLS, M. (org.). A sociedade em rede: do conhecimento à ação política. Lisboa: Imprensa Nacional, 2005. p. 18
- CHELLA, M. T. (2002). **Ambiente de robótica para aplicações educacionais com SuperLogo.** Universidade Estadual de Campinas, Campinas: São Paulo.
- CURY, T. E.; HIRSCHMANN, D. R. **Ensino de Matemática através do Arduino.** Porto Alegre. IERGS/UNIASSEL. VI, 2014.
- D'AMBROSIO, U. (1990). **Etnomatemática.** São Paulo: Ática.
- DELAZERI, H. W.; ISERHARDT, N. F.; LOPES, L. A.; ZONTA, E. F. **Desenvolvimento de um carrinho autônomo com Arduino para o ensino de Cinemática.** Instituto Federal de Educação Ciência e Tecnologia do Rio Grande do Sul, Canoas – 2015.
- EVANS, M.; NOBLE, J.; HOCHENBAUM, J. **Arduino em Ação.** Novatec Editora. São Paulo 2013.
- FAGUNDES, C. A. N.; POMPERMAYER, E. M.; BASSO, M. V. de A.; JARDIM, R. F. **Aprendendo Matemática com Robótica.** UFRGS: Porto Alegre, 2005. Instituto de Matemática, Universidade do Rio Grande do Sul.

GARLET, D.; BIGOLIN, N. M.; SILVEIRA, S. R. **Uma Proposta para o Ensino de Programação de Computadores na Educação Básica**, Departamento de Tecnologia da Informação, Universidade Federal de Santa Maria, RS, 2016.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4.ed. São Paulo: Atlas,2002.

GIL, A. C. **Métodos e técnicas de pesquisa social**. São Paulo: Atlas, 2008.

GODOY, A. S. **Introdução à pesquisa qualitativa e suas possibilidades**. RAE - Revista de Administração de Empresas, São Paulo, v. 35, n. 2, p. 57-63, 1995.

IEZZI, G., MURAKAMI, C. **Fundamentos de Matemática Elementar. Vol. 3**. 7ª ed. São Paulo: Atual Editora, 1993.

IMENES, L. M. P. (1989). **Um estudo sobre o fracasso do ensino e da aprendizagem da matemática**. Dissertação de Mestrado, UNESP. Rio Claro.

LAY, D. C. **Álgebra Linear e suas aplicações**. 2ª edição. Trad. Ricardo Camelier e Valéria de Magalhães Lório. Rio de Janeiro: LTC, 1999.

LIPSCHUTZ, S. **Álgebra Linear**. Trad. Roberto Ribeiro Baldino. São Paulo: McGraw-Hill, 1972.

MACEDO, R. S., PRIETCH, S. S. **Utilizando a Placa Arduino como Ferramenta de Apoio ao Ensino de Programação**, In IV Escola Regional de Informática da Sociedade Brasileira de Computação (SBC), 2013 – Regional de Mato Grosso.

MCROBERTS, M. **Arduino básico**. Tradução Rafael Zanolli. São Paulo: Novatec Editora, 2011.

MESQUITA, L. C. **Transformações ortogonais aplicada à cinemática de robôs**. 2016. 62f. Dissertação (Mestrado Profissional em Matemática em Rede Nacional) – Universidade Federal do Piauí, Teresina, 2016. Cap. 4.

NEVES, J. L. **Pesquisa qualitativa – características, uso e possibilidades**. Cadernos de pesquisa em administração, São Paulo. V. 1, nº 3, 2ºsem. 1996.

OLIVEIRA, A. S. **Sistemas Embarcados: hardware e o firmware na prática/ 1. ed**. São Paulo: Érica, 2006.

OLIVEIRA, E. S. **Um Breve Prognostico do uso da Robótica Educativa na prática educacional de discentes/professores do mestrado MECM-UEPB**. In: XI Encontro Nacional de Educação Matemática. Anais... Curitiba, 2013. Artigo, p.7991-7998. ISSN 2301-0797.

OLIVEIRA, G. A. A.; BETTIO, R. W.; RODARTE, A. P. M.; BRAZ, J. E.; FERRARI, F. B. **GrubiBots Educacional: jogo para o ensino de algoritmos na educação básica.** In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2014, Dourados.

PANCIERA, L. M.; FERREIRA, M. V. **A Modelagem Matemática no Ensino de Matrizes e Sistemas Lineares.** Unifra, RS, 2006.

PANTOJA, W. R. S. **Aplicação das operações com matrizes: Movimentando um robô no plano.** Dissertação (Mestrado Profissional em Matemática em Rede Nacional) – Universidade Federal Amapá- UNIFAP, Amapá, 2016, 42f.

RICHARDSON, R. J. **Pesquisa social: métodos e técnicas.** 3ª Ed. São Paulo: Editora Atlas, 2007.

SANCHES, M. H. F. **Efeitos de uma estratégia diferenciada do ensino dos conceitos de matrizes.** Dissertação (Mestrado em Educação), UNICAMP, Campinas, 2002.

SCHWARTZ, E. **Transformações lineares no processamento de imagens e na criptografia.** 2016. 38f. Dissertação (Trabalho de Conclusão de Curso) – Licenciatura em Matemática, Universidade Federal de São João Del-Rei, São João Del-Rei, 2016. Cap. 2.

SBC, Sociedade Brasileira de Computação (2016). **Computação na Base Nacional Comum Curricular.** Elaborada pela comissão de Educação da SBC em colaboração com a Comissão Especial de Informática na Educação e membros da SBC.

SCHWAB, K. **A Quarta Revolução Industrial.** Tradução: Daniel M. Miranda. Edipro, 2016.

SELIGER, H. W.; SHOHAMY, E. **Second Language Research Methods.** Oxford: Oxford University Press, 2001.

SETTI, M. O. G. O. **Processo de Discretização do Raciocínio Matemático na Tradução para o Raciocínio Computacional: Um Estudo de Caso no Ensino/Aprendizagem de Algoritmos.** Tese (Doutorado em Educação). Curitiba: UFPR, 2009.

SILVA, Y. G. **Controle supervisorio modular local de sistemas de veículos autoguiados.** Dissertação (Mestrado em Engenharia de Automação e Sistemas), Universidade Federal de Santa Catarina, Florianópolis, 2010.

SILVA, F. I.; SCHERER, D. **Praxedes: Protótipo de um kit educacional de robótica baseado na plataforma Arduino**. Revista EAD & Tecnologias Digitais na Educação. 2013.

SILVA, F. A. A. **Utilizando o Arduino como atividade aberta de investigação e experimentação matemática para o ensino de conceitos de matrizes**. Dissertação (Mestrado em Ensino de Ciências e Matemática) – Instituto federal de Educação, Ciência e Tecnologia de São Paulo. São Paulo, 2017. 161 f.

SOUZA, A. R. de *et al.* **A placa Arduino: uma opção de baixo custo para experiências de física assistidas pelo PC**. Rev. Bras. Ensino Fís. [online]. 2011, vol.33, n.1, pp.01-05. ISSN 1806-1117

STEINBRUCH, A; WINTERLE, P. **Álgebra Linear**. Ed. Person Makron Books Editora LTDA, 2. ed., São Paulo, 1987.

TAMASHIRO, R. M.; GANOKA, C. T.; CARDOSO, A. **Resenha: a quarta revolução industrial / Klaus Schwab; tradução Daniel Moreira Miranda**. Revista Ciências do Trabalho, on-line, nº9, dezembro, 2017. Disponível: <https://rct.dieese.org.br/index.php/rct/issue/view/10/showToc>. Acesso em 01 de junho de 2018. ISSN 2319-0574.

TRENTIN, M. A. S; PÉREZ, C. A. S; TEIXEIRA, A. C. **A robótica livre no auxílio da aprendizagem do movimento retilíneo**. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (CBIE), 2, 2013, Campinas.

TREVELIN, E. C. **Apostila Sobre a Linguagem C++ - Trabalho de Formatura Com Autoria de Enrique Camargo Trevelin**. Orientado pelo Prof. Dr. Carlos Antônio Alves. UNESP, 2007.

TRINDADE, A. G. **Apostila de Linguagem dev c ++**. Blog da Prof Andrea Garcia. 2010; Tema: Material de apoio para aulas da ETEC Adolpho Berezin. (Blog).

TORCATO, P. **O Robô ajuda? Estudo do Impacto do uso de Robótica Educativa como Estratégia de Aprendizagem na disciplina de aplicações informáticas B**. Congresso Internacional de TIC e Educação. Instituto de Educação da Universidade de Lisboa, 2012.

VILAÇA, M. L. C. **Pesquisa e ensino: considerações e reflexões**. Revista do Curso de Letras da UNIABEU Nilópolis, v. I, Número2, Mai. -Ago. 2010.

WING, J. M. (2006). **Computational thinking**. Communications of the ACM, v. 49, n. 3, p. 33–35..

Arduino. Site Oficial disponível em <https://www.Arduino.cc>. Acesso em fev. 2018

<https://blog.zerokol.com/2012/11/tcrt5000-reflective-optical-sensor.html>. Acesso em: jul. 2016

<http://osoyoo.com/>. Acesso em: Jul. 2018

<http://www.earduino.com.br>. Acesso em Jul. 2018

<https://www.theengineeringprojects.com>. Acesso em Jul. 2018

<http://www.huinfinito.com.br>. Acesso em Jul. 2018

<http://osoyoo.com/driver/smartcar-lesson4.zip>. Acesso em Jul. 2018

<https://www.imam.com.br/consultoria/artigos/arquivo-artigo>. Acesso em ago. 2018

<http://www.transliftbr.com/site/produtos-servicos/transportadores/sistema-agv>.

Acesso em: set. 2018

<http://www.newtoncbraga.com.br/index.php/robotica/5168-mec070a> acesso em set. 2018

<https://www.Arduinolandia.com.br>. Acesso em: set. 2018

<https://www.geogebra.org>. Acesso em nov. 2018