



UNIVERSIDADE FEDERAL DOS VALES DO JEQUITINHONHA E MUCURI
Programa de Pós-Graduação em Educação
Rafael Porto Viana

ROBÔBLOCKS: uma metodologia de ensino para facilitar o aprendizado de programação

Diamantina
2020

Rafael Porto Viana

ROBÔBLOCKS: uma metodologia de ensino para facilitar o aprendizado de programação

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Educação, como parte dos requisitos exigidos para a obtenção do título de Mestre em Educação.

Orientador: Alexandre Ramos Fonseca

Coorientador: Euler Guimarães Horta

Diamantina

2020

Elaborado com os dados fornecidos pelo (a) autor (a).

V614r

Viana, Rafael Porto

Robôblocks: uma metodologia de ensino para facilitar o
aprendizado de programação / Rafael Porto Viana, 2020.

76 p. : il.

Orientador: Alexandre Ramos Fonseca

Coorientador: Euler Guimarães Horta

Dissertação (Mestrado - Programa de Pós-Graduação em
Educação) - Universidade Federal dos Vales do Jequitinhonha e
Mucuri, Diamantina, 2020.

1. Robótica. 2. Kit educacional. 3. Ensino de programação. I.
Fonseca, Alexandre Ramos. II. Horta, Euler Guimarães. III. Título. VI.
Universidade Federal dos Vales do Jequitinhonha e Mucuri.

CDD 370.3078

Ficha Catalográfica – Sistema de Bibliotecas/UFVJM

Bibliotecária: Viviane Pedrosa – CRB6/2641



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DOS VALES DO JEQUITINHONHA E MUCURI

RAFAEL PORTO VIANA

ROBÔBLOCKS: uma metodologia de ensino para facilitar o aprendizado de programação

Dissertação apresentada ao Programa de Pós-Graduação em Educação da Universidade Federal dos Vales do Jequitinhonha e Mucuri, **nível de Mestrado**, como requisito parcial para obtenção do título de **Mestre em Educação**.

Orientador: Prof. **Alexandre Ramos Fonseca**

Data de aprovação 01/12/2020.

Prof. Alexandre Ramos Fonseca - (UFVJM)

Prof. Euler Guimarães Horta - (UFVJM)

Prof. Maria Lúcia Bento Villela - (UFVJM)

Prof. Renato Dourado Maia - (UFMG)



Documento assinado eletronicamente por **Alexandre Ramos Fonseca, Servidor**, em 01/12/2020, às 15:28, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Euler Guimaraes Horta, Servidor**, em 01/12/2020, às 15:28, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Maria Lucia Bento Villela, Servidor**, em 01/12/2020, às 15:28, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Renato Dourado Maia, Usuário Externo**, em 01/12/2020, às 15:29, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufvjm.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0229854** e o código CRC **3609A695**.

*Dedico este trabalho aos meus pais, **Zelito e Sandra**,
que sempre apoiaram minhas decisões e incentivaram meus sonhos.
Ao meu irmão **Rodrigo** (in memoriam) e à minha irmã **Mariana**,
por sempre acreditarem em mim.
Aos meus avós **Antenor** (in memoriam), **Olíbia, José Porto e Elisa**,
que me ensinaram valores importantes para toda a vida.
À minha amada esposa **Janine**, ao meu filho **Heitor** e às minhas filhas **Isis e Yasmim**, por todo
amor, incentivo, apoio e compreensão. Nada disso teria sentido se vocês não existissem na
minha vida.*

AGRADECIMENTOS

Agradeço primeiramente a Deus, por ser o meu maior orientador e nunca me abandonar nos momentos de necessidade.

Aos meus orientadores, Prof. Alexandre Ramos Fonseca e Prof. Euler Guimarães Horta, pelo companheirismo, pela confiança em mim depositada e pela parceria constante na realização desse trabalho.

À professora Dra. Maria Lúcia Bento Villela e ao professor Dr. Renato Dourado Maia pela participação na minha banca de qualificação e defesa, que muito contribuíram com sugestões e melhorias desde o primeiro momento.

Aos colegas de mestrado da UFVJM, especialmente a turma da biblioteca, que se abriram de boa vontade para a troca de experiências e deram todo o apoio para que este trabalho pudesse ser realizado.

Aos meus colegas Fábio Rodrigues e Daniel Bulhões, por nossas reuniões, sejam sobre nossos projetos de pesquisa ou sobre coisas da vida.

À direção, coordenação e alunos da 1ª série do curso Técnico em Informática integrado ao Ensino Médio do IFNMG - *Campus* Montes Claros, que participaram com interesse e entusiasmo deste trabalho.

Aos amigos do IFNMG - Campus Almenara, especialmente aos professores Alan Teixeira de Oliveira, Marcos Vinícius Montanari e Fernando Silvera Goulart Júnior (in memoriam), pelo total apoio, disponibilidade, pelos saberes que transmitiram de bom grado, pelas opiniões e críticas que me fizeram crescer e por todas as palavras de incentivo.

À minha prima Kamila, por sua preocupação, apoio e incentivo.

Ao meu primo Djiacono e sua esposa Priscila, pelo suporte, hospitalidade e por fazerem com que me sentisse sempre bem confortável.

Ao criador do DuinoBlocks4Kids, Rubens Lacerda, pelo pronto e gentil atendimento à todas as minhas solicitações de suporte.

Ao Programa de Bolsas para Qualificação de Servidores (PBQS) ofertado pelo Instituto Federal do Norte de Minas Gerais (IFNMG), do qual fui bolsista.

E, por fim, a minha família, que desde sempre deu todo suporte necessário aos meus projetos e sonhos, sendo meu principal alicerce, não somente nessa como em todas as minhas empreitadas.

RESUMO

A robótica vem se destacando por permitir o desenvolvimento de projetos educacionais por meio da construção e manipulação de robôs, em um ambiente de aprendizagem atrativo e dinâmico. Esta pesquisa tem como objetivo avaliar se uma metodologia de ensino que faz uso de um *Kit* de robótica educacional de baixo custo ameniza as dificuldades apresentadas por alunos iniciantes em programação. A metodologia de ensino e o *Kit* foram desenvolvidos ao longo deste trabalho, e aplicados aos estudantes da disciplina de Introdução à Programação, ofertada no primeiro ano do curso Técnico em Informática Integrado ao Ensino Médio, do IFNMG - *Campus* Montes Claros. A metodologia de ensino é dividida em três etapas, que devem ser executadas de forma sequencial. A primeira etapa envolve atividades sem o uso de equipamento eletrônico, enquanto a segunda e a terceira etapa são feitas com uso de computador e equipamento robótico. A opção metodológica deste projeto de pesquisa fundamenta-se na abordagem qualitativa e utiliza procedimentos técnicos da pesquisa de campo, visando a descrever os efeitos produzidos pela metodologia de ensino e pelo *kit* de robótica educacional. A avaliação da metodologia de ensino foi realizada a partir dos dados coletados por meio da observação do professor e do questionário aplicado aos estudantes. Embasando-se nos resultados obtidos, pode-se dizer que o objetivo da pesquisa, ainda que com poucos participantes, foi alcançado, visto que um *kit* de robótica educacional de baixo custo mostrou-se eficaz em amenizar as dificuldades apresentadas por alunos iniciantes em programação. A proposta teve uma boa aceitação, pois utiliza diferentes ferramentas tecnológicas que podem contribuir para o dinamismo das aulas, dando significado aos conteúdos apresentados e desenvolvendo o pensamento computacional.

Palavras-chave: Robótica. *Kit* Educacional. Ensino de Programação.

ABSTRACT

Robotics has stood out for allowing the development of educational projects through the construction and manipulation of robots in an attractive and dynamic learning environment. This research aims to evaluate whether a teaching methodology that uses a low cost educational robotics kit eases the difficulties presented by students who are new to programming. The teaching methodology and the Kit were developed throughout this work, and applied to students of the Introduction to Programming discipline, offered in the first year of the Technical Course in Informatics Integrated to High School, at the IFNMG Campus Montes Claros. The teaching methodology is divided into three stages, which must be carried out sequentially. The first stage involves activities without the use of electronic equipment, while the second and third stages are performed using a computer and robotic equipment. The methodological option of this research project is based on the qualitative approach and uses technical procedures from field research, aiming to describe the effects produced by the teaching methodology and the educational robotics kit. The evaluation of the teaching methodology was carried out based on the data collected through the observation of the teacher and the questionnaire applied to the students. Based on the results obtained, it can be said that the research objective, although with few participants, was achieved, since a low-cost educational robotics kit proved to be effective in easing the difficulties presented by students beginning in programming. The proposal was well accepted, as it uses different technological tools that can contribute to the dynamism of the classes, giving meaning to the content presented and developing computational thinking.

Keywords: Robotics. Educational Kit. Programming Teaching

LISTA DE ILUSTRAÇÕES

Figura 1 – Plataforma <i>Code.org</i>	25
Figura 2 – Interface do DuinoBlocks4Kids (DB4K)	28
Figura 3 – Robô seguindo linha	35
Figura 4 – Peças do Tabuleiro	36
Figura 5 – Movimentos do robô para seguir a linha.	36
Figura 6 – Reconhecimento extra	37
Figura 7 – Placa de prototipagem	37
Figura 8 – <i>Kit</i> Chassi Redondo 2WD Robô	38
Figura 9 – Sensor de linha infravermelho	38
Figura 10 – Módulo Ponte H L298N	39
Figura 11 – Disposição dos componentes no chassi	40
Figura 12 – Bloco do sensor de refletância	40
Figura 13 – Bloco da buzina	41
Figura 14 – Avançar com o buzzer ligado quando reconhecer o verde nos sensores da direita e esquerda	41
Figura 15 – Avançar quando reconhecer o preto nos sensores da direita e da esquerda	42
Figura 16 – Virar a direita quando reconhecer o branco no sensor da direita e o preto no sensor da esquerda	42
Figura 17 – Virar a esquerda quando reconhecer o branco no sensor da esquerda e o preto no sensor da direita	43
Figura 18 – Parar quando reconhecer o branco nos sensores da direita e da esquerda	43
Figura 19 – Atividade 1 - Computação Desplugada	47
Figura 20 – Atividade 2 - Programação em Blocos no CODE.org	48
Figura 21 – Atividade 3 - Programação em Blocos para Arduino no DB4K	48
Figura 22 – Atividade 3 - Montagem do percurso	49
Figura 23 – Atividade 3 - Testando a Programação	49
Gráfico 1 – Você teve contato com alguma linguagem de programação antes de entrar no curso?	50
Gráfico 2 – Quais linguagens de programação você costumava utilizar?	50
Gráfico 3 – Metodologia de ensino RobôBlocks	51
Gráfico 4 – Você acredita que obteve conhecimentos importantes para desenvolver um programa?	52
Gráfico 5 – Habilidades e Competências estimuladas por meio do <i>Kit</i> Educacional RobôBlocks	52
Gráfico 6 – Habilidades e Competências estimuladas por meio do <i>Kit</i> Educacional RobôBlocks	53
Gráfico 7 – A metodologia de ensino proposta para aplicação do <i>kit</i> possui as seguintes características?	53

Gráfico 8 – Distribuição das atividades da oficina sugerida 54

LISTA DE TABELAS

Tabela 1 – Custos para montagem do *Kit* 44

LISTA DE ABREVIATURAS E SIGLAS

CEP	Comitê de Ética em Pesquisa
CSTA	<i>Computer Science Teachers Association</i>
CONEP	Comissão Nacional de Ética em Pesquisa
DB4K	DuinoBlocs4Kids
DC	<i>Direct Current</i>
IFNMG	Instituto Federal do Norte de Minas Gerais
IR	<i>Infra-Red</i>
ISTE	<i>International Society for Technology in Education</i>
LED	<i>Light-Emitting Diode</i>
MEO	<i>Ministry of Education Ontario</i>
MEC	Ministério da Educação
MIT	<i>Instituto de Tecnologia de Massachusetts</i>
NSF	<i>National Science Foundation</i>
PWM	<i>Pulse Width Modulation</i>
TAM	Termo de Assentimento Para Menor
TCLE	Termo de Consentimento Livre e Esclarecido
UFVJM	Universidade Federal dos Vales do Jequitinhonha e Mucuri

SUMÁRIO

1	INTRODUÇÃO	21
2	REVISÃO DE LITERATURA	23
2.1	Contribuições do Pensamento Computacional na Educação	23
2.2	Programação Visual	24
2.3	Robótica Educacional	26
2.3.1	<i>Arduino</i>	28
2.3.2	<i>Duinoblock4kids</i>	28
2.4	Trabalhos Relacionados	29
3	MATERIAIS E MÉTODOS	33
3.1	Ambiente de estudo	33
3.2	Metodologia de Ensino	34
3.3	Kit Educacional Robôblocks	35
3.3.1	<i>Como o robô (carrinho) seguidor de linha funciona?</i>	36
3.3.2	<i>Componentes e Montagem do Kit</i>	37
3.3.2.1	<i>Sensor de linha infravermelho Tcrt5000</i>	38
3.3.2.2	<i>Ponte H L298</i>	39
3.4	Software	40
3.5	Testes Realizados com o Kit RobôBlocks	41
3.6	Custos para montagem do Kit Robôblocks	44
3.7	Coletas de dados	45
4	RESULTADOS E DISCUSSÕES	47
5	CONSIDERAÇÕES FINAIS	55
	REFERÊNCIAS	57
	APÊNDICE A – ATIVIDADE DESPLUGADA	63
	APÊNDICE B – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO E QUESTIONÁRIO	71
	ANEXO A – MANUAL DE INSTRUÇÕES	75

1 INTRODUÇÃO

Os avanços tecnológicos permitem adotar métodos mais modernos de ensino, ao provocar mudanças no modelo tradicional de aprendizagem, uma vez que objetos antigos do cotidiano escolar, como giz e quadro, perdem espaço para computadores e outras tecnologias. Leite *et al.* (2000) defendem a utilização desses recursos em sala de aula, acreditando que a modernização no universo escolar, além de contribuir para o desenvolvimento do aluno, tende a torná-lo mais crítico e atualizado.

O pensamento crítico associado a noções básicas da computação caracteriza um procedimento para resolução de problemas, que Wing (2006) descreveu como Pensamento Computacional. Essa mesma autora defende a introdução do pensar computacional nos anos iniciais e o considera uma habilidade intelectual básica do ser humano, relevante para solução de problemas do mundo atual.

Segundo Gallo (2000 apud MARTINS; OLIVEIRA; OLIVEIRA, 2012), educar cidadãos para os desafios do mundo contemporâneo requer reinventar as práticas curriculares, de modo que os conhecimentos não fiquem isolados. Nessa perspectiva de transformações no modo de pensar as práticas de ensino e aprendizagem, a robótica educacional tem se demonstrado um ambiente pedagógico enriquecedor (SANTOS; MENEZES, 2005). Exemplo disso é o desenvolvimento de atividades por meio da robótica que, ao exigir dos estudantes a organização lógica de informações e pensamentos, servem para estimular e evidenciar indícios de aprendizagem (SILVA; JAVARONI, 2018).

Silva (2009) traz considerações mostrando que Seymour Papert, pesquisador do MIT (Instituto de Tecnologia de Massachusetts), foi o grande precursor da utilização de robôs como mediador para construção do conhecimento nos anos 60, quando também nascia o construcionismo. Nas palavras de Cambruzzi e Souza (2014, p.42), “a Robótica Educacional exige do aluno a organização de tarefas e pensamentos, desde o planejamento, até a montagem mecânica e a programação da lógica do robô”. De acordo com as observações dos autores, no decorrer do projeto é necessário agregar conhecimentos interdisciplinares com intuito de solucionar problemas, elevando gradualmente o pensamento computacional e a aproximação dos alunos.

Cambruzzi e Souza (2014) consideram que o estudo de condicionais e laços de repetição é complexo e exige um alto grau de abstração por parte dos alunos. Entretanto, ao materializar a programação em ações dos robôs, conceitos antes abstratos tornam-se ações físicas, que comprovam a lógica do pensamento. Um estudo realizado por Costa e Jacon (2019) constatou que a robótica educacional pode ser uma estratégia interessante para minimizar a problemática de evasão e retenção de disciplinas introdutórias de programação.

O grau de complexidade nos cursos de programação faz com que ocorram altos índices de reprovação e evasão, que prejudicam tanto a instituição formadora, que acaba demandando um recurso maior para manter o aluno até formar um profissional, quanto o mercado de trabalho, que aguarda mais tempo por esse profissional (BULHÕES *et al.*, 2019). Sendo assim, fica notória a necessidade de procurar alternativas para melhorar o aprendizado, ampliando

os métodos de ensino, de modo que esses estejam adequados à realidade atual, com o intuito principal de formar indivíduos mais capacitados para o mercado de trabalho.

Para tanto, este trabalho tem como objetivo geral avaliar se uma metodologia de ensino que faz uso de um *Kit* de robótica educacional de baixo custo ameniza as dificuldades apresentadas por alunos iniciantes em programação. A metodologia e o *Kit* foram desenvolvidos ao longo deste trabalho e podem ser aplicados a estudantes de qualquer área do ensino que não envolva a computação, visto que destina-se ao desenvolvimento do pensamento computacional.

Os autores Cavalcante, Costa e Araujo (2016) salientam que a inserção do pensamento computacional aos estudantes é uma metodologia que vem sendo bastante pesquisada e discutida como uma forma de aprimorar o raciocínio lógico em diversas áreas do ensino. Dessa forma, espera-se que este estudo possa contribuir para a discussão de novas metodologias de ensino, destacando a necessidade de desenvolver o pensamento computacional e ampliar os métodos de aprendizagem, de modo que estejam adequados à realidade atual, com o intuito de formar indivíduos mais capacitados para o mercado de trabalho.

Este texto está dividido em cinco capítulos e segue organizado da seguinte maneira: o primeiro é uma introdução para contextualização do trabalho; o segundo é uma revisão de literatura que relata algumas pesquisas e ferramentas lúdicas relacionadas ao ensino de programação; o terceiro descreve os materiais e métodos utilizados neste trabalho; o quarto aborda os resultados e discussões obtidos; e o quinto apresenta as considerações finais.

2 REVISÃO DE LITERATURA

Esta seção apresenta a importância em ensinar programação para iniciantes, por meio da utilização de ferramentas lúdicas, como a programação visual e a robótica educacional. Essas ferramentas têm o intuito de contribuir para o êxito da aprendizagem dos alunos, ao mesmo tempo em que tende a diminuir a evasão escolar, pois os estudantes se sentem seduzidos por esses instrumentos tecnológicos.

2.1 Contribuições do Pensamento Computacional na Educação

Desde a década de 70, Papert (1971) já observava que a programação de computadores aprimorava o pensamento e propiciava um avanço no processo de construção do conhecimento. O autor ainda ressalta que a computação materializa e esclarece concepções amenas em diversas áreas de conhecimento, aproximando a imaginação da realidade, o que provoca um aumento significativo no aprendizado.

A sociedade e suas diversas necessidades, de certa forma, estão atreladas à computação, sendo difícil imaginar a ausência de tecnologias e computadores atualmente (FRANÇA *et al.*, 2014). Nesse sentido, o pensamento crítico associado a noções básicas da computação caracteriza um procedimento para resolução de problemas, que Wing (2006) descreveu como Pensamento Computacional. A autora ainda defende a introdução do pensar computacional nos anos iniciais do ensino básico, despertando assim o interesse dos alunos, mostrando a versatilidade e relevância na solução de problemas do mundo atual.

Alguns países vêm introduzindo no currículo mínimo da educação básica o ensino de Ciências da Computação, como o Canadá (MEO, 2008) e os Estados Unidos, que investem na educação por considerarem que mais significativo do que aprender temas relacionados à tecnologia é a habilidade de potencializar nos alunos o pensamento computacional e a autossuficiência na solução de problemas. Exemplo clássico é a Computação Desplugada, um projeto conhecido por proporcionar o desenvolvimento do pensamento computacional, mesmo que o aluno esteja em condições inadequadas e não faça uso de computador (SCAICO *et al.*, 2012).

Um consenso entre vários autores é que desenvolver o Pensamento Computacional não está relacionado a fazer com que as pessoas pensem como cientistas da computação, mas provocar nelas o desejo de buscar novos conhecimentos para solucionar diversos problemas por meio de uma maneira específica de pensar, utilizando habilidades necessárias para o desenvolvimento de programas computacionais (ANDRADE *et al.*, 2013; BARR; STEPHENSON, 2011; HEMMENDINGER, 2010).

Andrade *et al.* (2013) consideram que três fundamentos básicos sustentam o Pensamento Computacional: a abstração – solucionar problemas a partir da extração de características do imaginário; a automação – utilizar tecnologia para substituir trabalhos manuais; e a análise – estudar os resultados gerados pela automação de processo. Na opinião de Blikstein (2008), o Pensamento Computacional é o aumento do poder cognitivo e operacional através da utilização do computador. Para tanto, esse autor aponta duas etapas para o pensar computacionalmente:

definir tarefas que podem ser desenvolvidas por um computador de forma eficiente e rápida; e saber programar essas tarefas para serem executadas por um computador.

As organizações CSTA (*Computer Science Teachers Association*), ISTE (*International Society for Technology in Education*) e a NSF (*National Science Foundation*) acreditam que todos os alunos devem dominar habilidades básicas do Pensamento Computacional na conclusão do ensino médio, podendo ser estimuladas por conceitos fundamentais como: coleta, análise e representação de dados; abstração e decomposição de problemas; automação dos procedimentos; simulação e paralelismo (CSTA, 2011). O estímulo de habilidades do Pensamento Computacional proporciona o envolvimento de alunos e professores em projetos interdisciplinares, nos quais a Computação atua como meio para solução de problemas, além de existir a possibilidade de criar ferramentas através de ambientes de programação, com o objetivo de apoiar o ensino de diversas áreas (FRANÇA *et al.*, 2014).

Apesar da importância do desenvolvimento do Pensamento Computacional em crianças e jovens, a assimilação de conceitos a ele relacionados nem sempre é trivial, principalmente aqueles relacionados à programação de computadores. Assim, diante das dificuldades apresentadas por iniciantes na aprendizagem de programação, foram desenvolvidos ambientes visuais de programação. Essa ideia utiliza blocos de código em componentes visuais, de forma a simplificar e facilitar o ensino da programação, sendo necessário que o aluno foque apenas na lógica de funcionamento dos blocos, sem se preocupar com a complexa sintaxe de uma linguagem de programação (FRANÇA *et al.*, 2014).

2.2 Programação Visual

Ao ensinar programação é comum utilizar o pseudocódigo, por possuir uma linguagem simples, permitindo que o aluno foque a princípio na solução do problema em questão, sem ter que conhecer a sintaxe de uma linguagem de programação (SOUZA, 2009). Com base nas ideias de Smith (2000), observou-se que os alunos que estudam programação utilizando o método por demonstração tendem a adquirir melhores rendimentos do que aqueles que aprenderam pela maneira tradicional. A programação por demonstração é um método que permite que os usuários, por intermédio de um ambiente visual de programação, aprendam a programar sem se preocupar com “como” vai ser feito, mas sim, com “o que” deve ser feito (FERREIRA; GONZAGA; SANTOS, 2010).

Segundo Evangelista (2001), enquanto a linguagem textual demanda do estudante o entendimento das particularidades de cada linguagem de programação que pretende manipular, a linguagem visual aborda somente a lógica de desenvolvimento do algoritmo, removendo as dificuldades da sintaxe e focando na semântica do programa. Além disso, o autor ressalta que o entendimento de um problema através da representação visual é muito maior se comparada à representação textual, facilitando para se obter uma solução.

Existe uma discussão teórica sobre quais os melhores métodos para ensinar programação e alcançar as competências básicas. Porém, até então não existe um entendimento sobre o melhor método para ensinar e avaliar durante a educação básica. Algumas pesquisas defen-

dem as linguagens visuais, como o *Scratch* (SCAICO *et al.*, 2012; AURELIANO; TEDESCO, 2012; ARAÚJO *et al.*, 2013) e plataformas que acompanhem essa linha, como o App Inventor (GOMES; MELO, 2013).

Nesta perspectiva, a plataforma *Code.org* é um ambiente de programação que disponibiliza vários jogos destinados a trabalhar os conceitos de lógica de programação. Esses jogos são agrupados em cursos. Existem cursos mais fáceis, voltados para crianças ainda em fase de alfabetização, bem como cursos com dificuldade média, voltados para crianças da educação básica, e cursos avançados, voltados para jovens do ensino médio. Gramigna (2007) explica que o jogo é um método importante na educação, por envolver as ações de criar e implementar soluções de forma lúdica, estimulando características essenciais para o desenvolvimento por completo, entre elas, valores éticos, autodisciplina, sociabilidade, espírito de equipe e sensatez.

O ambiente oferece para os educadores mecanismos para criação de turmas. Nessas turmas é possível cadastrar os alunos e acompanhar seu progresso. O professor tem acesso a todas as atividades realizadas pelos alunos e consegue, de forma simples, visualizar os pontos em que os alunos encontram dificuldade.

Na Figura 1 é apresentado o curso 2 da plataforma *Code.org*. Nesse curso, os alunos vão criar programas para resolver problemas e desenvolver jogos ou histórias interativas que eles podem compartilhar.

Figura 1 – Plataforma *Code.org*



Fonte: CODE, 2013.

O curso começa apresentando um labirinto simples onde o aluno pode andar com o personagem para frente e virar à esquerda ou à direita. Nessa etapa, é ensinado o conceito de comandos. À medida em que as fases vão sendo vencidas pelo estudante, o nível de dificuldade vai aumentando e novos conceitos são apresentados. Esses conceitos são ferramentas de programação que auxiliam na resolução de problemas mais complexos de uma forma mais simples. Por exemplo: se é necessário andar para a frente 10 vezes, pode-se, num primeiro momento, usar 10 blocos “avance”, mas se utilizado o conceito de repetição, pode-se usar um bloco “repetir 10x” e um bloco “avance”.

Todos os cursos disponibilizados pela plataforma *Code.org* contam com planos de ensino, vídeo-aulas explicativas, ferramentas de avaliação de desempenho e permitem a emissão

de certificados no formato digital para os alunos que concluírem as atividades. Um ponto interessante é que algumas das vídeo-aulas são apresentadas por personalidades famosas no mundo da computação, como Bill Gates (cofundador da Microsoft), Mark Zuckerberg (cofundador do Facebook), além de desenvolvedores da Disney, entre outros. Isso motiva os participantes, pois mostram pessoas de destaque discutindo a importância de se aprender lógica de programação.

2.3 Robótica Educacional

A utilização de tecnologias está cada vez mais presente no mundo moderno, facilitando cada vez mais o trabalho do homem. Papert (1994), o precursor da robótica Educacional, afirma que existe um paradoxo em relação ao uso da tecnologia na educação, pois acarreta uma mudança que virá através da utilização de meios técnicos para eliminar a natureza técnica da aprendizagem na escola. Ele afirma ainda que, através do uso das tecnologias, é possível inovar métodos e técnicas do professor, ampliando as possibilidades de aprendizagem.

Assim como as outras tecnologias, a robótica vem sendo utilizada na educação como elemento facilitador no processo educativo. Mas o que realmente é robótica? Segundo César e Bonilla (2007), robótica é a ciência dos sistemas que interagem com o mundo real com ou sem intervenção dos humanos. Ela está em expansão e é considerada multidisciplinar. Em outra perspectiva, Silva (2009) afirma que a robótica estimula uma nova transformação dos meios de produção, devido à sua multifuncionalidade automática, que é resultante da leitura, interpretação e tratamento de informações do seu ambiente de trabalho.

Santos e Menezes (2005, p.2), destacam que a Robótica Educacional pode ser definida como “um ambiente onde o aprendiz tenha acesso a computadores, componentes eletromecânicos (motores, engrenagens, sensores, rodas etc.), eletrônicos (interface de *hardware*) e um ambiente de programação para que esses componentes possam funcionar”. Em outras palavras, refere-se à implementação de robôs com uso de dispositivos interfaceados com finalidades educacionais. Não obstante, Júnior, Vasques e Francisco (2010), salientam que a robótica, quando inserida no ambiente escolar, é capaz de criar laços entre os alunos, deixando de lado suas diferenças, promovendo relações na criação das ideias, no desenvolvimento e na execução dos projetos.

Nesse sentido, Maisonnette (2002) define a Robótica Educativa como sendo o controle de mecanismos eletro-eletrônicos através de um computador, transformando-o em uma máquina capaz de interagir com o meio ambiente e executar ações definidas por um programa criado pelo programador a partir dessas interações. O autor ainda ressalta o potencial da robótica como ferramenta interdisciplinar, tendo em vista que a construção de um novo mecanismo, ou a solução de um novo problema frequentemente extrapola a sala de aula. Devido a isso, o aluno questiona professores de outras disciplinas, na tentativa de buscar respostas para a solução do seu problema.

Ainda nesse enfoque, Maisonnette (2002) afirma que, com a robótica educacional, o aluno passa a construir seu conhecimento através de suas próprias observações, e aquilo que é aprendido pelo esforço próprio da criança tem muito mais significado para ela e se adapta às

suas estruturas mentais. O referido autor menciona que esse novo recurso permite que haja a integração de diversas disciplinas e a simulação do método científico, pois o aluno formula uma hipótese, implementa, testa, observa e faz as devidas alterações para que o seu “robô” funcione.

Na palavras de Zilli (2002), a Robótica Educacional, além de propiciar ao educando o conhecimento da tecnologia atual, o ajuda a desenvolver: a capacidade de raciocínio lógico; habilidades manuais e estéticas; relações interpessoais e intrapessoais; utilização de conceitos aprendidos em diversas áreas do conhecimento para o desenvolvimento de projetos; investigação e compreensão; representação e comunicação; trabalho com pesquisa; resolução de problemas por meio de erros e acertos; aplicação das teorias formuladas a atividades concretas; utilização da criatividade em diferentes situações; capacidade crítica.

César (2010) considera que o objetivo da robótica educacional é desenvolver projetos educacionais por meio da construção e manipulação de robôs, visando a proporcionar aos alunos um ambiente de aprendizagem que possibilite o desenvolvimento do raciocínio, criatividade, conhecimento multidisciplinar, bem como prepará-los para o mundo atual, já que cada vez mais se faz necessário a interação homem-máquina para a realização das tarefas diárias. Esse mesmo autor salienta que, de forma geral, as principais vantagens da robótica educativa são:

- Desenvolver a auto-suficiência na busca e obtenção de conhecimentos;
- Permitir testar em um equipamento físico o que aprenderam utilizando modelos que simulam o mundo real;
- Estimular a leitura, a exploração e a investigação;
- Preparar o aluno para trabalho em grupo;
- Estimular o hábito do trabalho organizado, uma vez que desenvolve aspectos ligados ao planejamento, execução e avaliação final de projetos;
- Ajudar na superação de limitações de comunicação, fazendo com que o aluno verbalize seus conhecimentos e suas experiências e desenvolva sua capacidade de argumentar e contra-argumentar;
- Desenvolver concentração, disciplina, responsabilidade, persistência e perseverança;
- Aprimorar a motricidade, através da execução de trabalhos manuais;
- Estimular a criatividade, tanto no momento de concepção das ideias, como durante o processo de resolução dos problemas;
- Desenvolver o raciocínio e a lógica na construção de algoritmos e programas para controle de mecanismos;
- Favorecer a interdisciplinaridade, promovendo a integração de conceitos de áreas como: matemática, física, eletricidade, eletrônica, mecânica e arquitetura.

Pode-se concluir que a robótica educacional visa a desenvolver o conhecimento do aluno transformando a teoria em prática, com o intuito de utilizar conceitos obtidos em sala de aula e no cotidiano, para desenvolver habilidades capazes de organizar e solucionar problemas (MELO; AZOUBEL; PADILHA, 2009).

2.3.1 Arduino

O Arduino foi projetado por um grupo de cinco pesquisadores de distintas nacionalidades, no *Interaction Design Institute Ivrea42*, situado na cidade de Ivrea, Itália. O objetivo era envolver *software* e *hardware* livre para prototipação de projetos interativos, de forma a simplificar a utilização por qualquer pessoa, de modo que possam facilmente desenvolver produtos, mesmo sem ter um conhecimento avançado de eletrônica (BANZI, 2012).

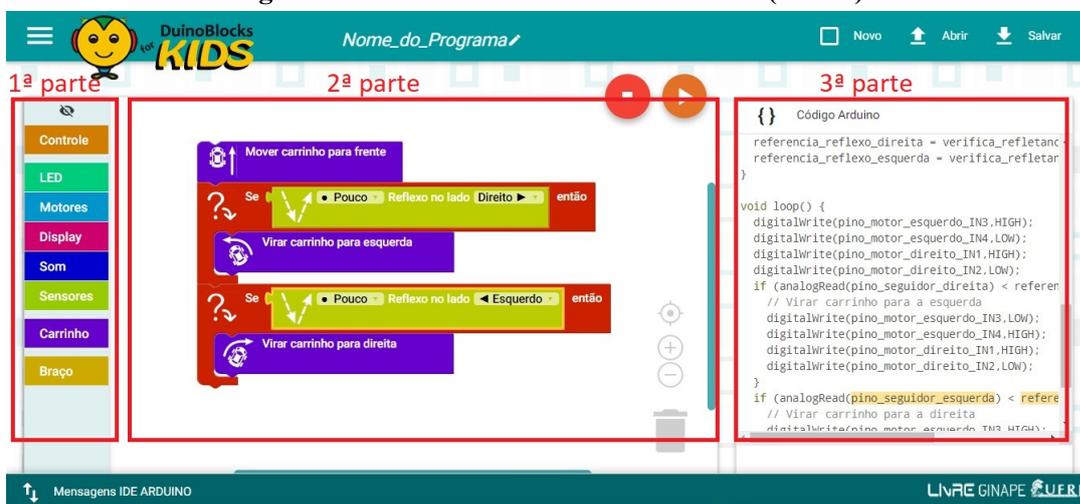
No Arduino, todos os esquemas dos circuitos eletrônicos, as documentações e metodologias de ensino estão disponíveis para *download* e podem ser utilizadas livremente, isso porque a *hardware* possui o código aberto e somente o nome da marca “Arduino” é protegido (BANZI, 2012). As placas de prototipagem eletrônica, como as do Arduino, têm proporcionado o desenvolvimento de diversos projetos relevantes. O seu baixo custo, versatilidade e facilidade de uso formam uma ótima opção para o desenvolvimento de projetos na área da Robótica Educacional.

2.3.2 Duinoblock4kids

O Laboratório de Inovações em Robótica Educacional (LIVRE) vem desenvolvendo, desde 2011, softwares e hardwares visando abarcar métodos de ensino que facilitem e ampliem o trabalho de professores e alunos em novos contextos de ensino com tecnologias. A exemplo disso, o DuinoBlocks4Kids (DB4K) é uma plataforma para o aprendizado de programação por meio da Robótica Educacional, destinado a alunos do Ensino Fundamental, totalmente baseada em Tecnologia Livre (Código fonte aberto), composta por um ambiente de programação em blocos para placas de prototipagem eletrônica (LIVRE UFRJ, 2019).

O DB4K é uma ferramenta baseada na plataforma DuinoBlokcs, que possibilita a criação de código para Arduino, utilizando programação em blocos (Figura 2). O DuinoBlokcs, por sua vez, é um ambiente de programação gráfica baseado no Ardublockly e utiliza as bibliotecas Blockly e Materialize (LIVRE UFRJ, 2019).

Figura 2 – Interface do DuinoBlocks4Kids (DB4K)



O DB4K possui uma interface simples, que é dividida em três partes, conforme é ilustrado na Figura 2: a primeira permite ao usuário criar o código da aplicação utilizando elementos gráficos (menu de blocos); na segunda, o usuário pode definir o comportamento dos componentes escolhidos, de forma semelhante à utilizada no Code.org; na terceira é exibido o código Arduino gerado pelos blocos. Dessa forma, o aluno poderá materializar o conhecimento adquiridos no decorrer do projeto (LIVRE UFRJ, 2019).

Diante de todos os argumentos aqui expostos, fica evidenciada a relevância deste projeto que se propõe a analisar a utilização da robótica como tecnologia facilitadora no processo de ensino-aprendizagem de programação. Para esta pesquisa serão utilizados o CODE.org e o DuinoBlokcs4Kids (DB4K), com intenção de compor um *kit* educacional de baixo custo, que possibilite a criação de códigos, através da programação em blocos, deixando simples e prazerosa a construção de equipamentos robóticos por meio do Arduino.

2.4 Trabalhos Relacionados

Existem, na literatura, várias iniciativas educacionais que buscam o desenvolvimento do Pensamento Computacional, em diferentes níveis de ensino, por meio de diversas técnicas, conforme se apresenta:

a) Trabalhos que usam Computação Desplugada

Bulhões *et al.* (2019) executaram um quase-experimento, utilizando técnicas de computação desplugada, com alunos iniciantes de dois cursos técnicos em informática. Os resultados mostraram um aumento do rendimento dos estudantes em relação ao momento que iniciaram o curso, além de apresentar melhores resultados em relação às técnicas utilizadas habitualmente.

b) Trabalhos que fazem uso da Programação Visual

Ramos e Teixeira (2015) fizeram uma análise com alunos de turmas do Ensino Médio de uma escola pública, e constataram que mesmo sem conhecimentos prévios específicos da área da computação, e ainda num espaço reduzido de tempo, os alunos tiveram um desenvolvimento significativo do Pensamento Computacional por meio do Scratch. As autoras Gomes e Melo (2013) apontam a metodologia como estratégia de ensino de programação no relato de experiência produzido, após apresentar e aplicar uma proposta por meio do ambiente de programação visual.

Ao investigar o ambiente de programação visual, os pesquisadores Aureliano e Tedesco (2012) acordaram que o uso da técnica proporcionou melhores resultados aos alunos que estão iniciando os estudos, comparado à linguagem textual, mas reconhecem que é preciso mais investigação sobre o tema. Scaico *et al.* (2012) também trazem seus pareceres descrevendo um relato de experiência no ensino de programação através de blocos códigos, ao relatar ser uma prática que estimula a criatividade e motiva os alunos.

c) Trabalhos que utilizam Robótica Educacional

No que diz respeito a robótica educacional, Papparidis e Franco (2016) propuseram a utilização do Arduino para demonstrar conceitos básicos de programação a alunos de nível técnico. Os resultados se mostraram satisfatórios e constataram a capacidade de potencializar

o ensino de programação não somente no ensino técnico integrado, mas também no ensino superior, com uma possível minimização das taxas de reprovação e evasão.

Queiroz, Sampaio e Santos (2019) apresentam um estudo de caso sobre o desenvolvimento do Pensamento Computacional em crianças do Ensino Fundamental I, por meio da Robótica Educacional de baixo custo. Os resultados observados indicam a possibilidade de desenvolvimento das seguintes habilidades do Pensamento Computacional: capacidade de abstração, decomposição de problemas, compreensão de fluxos de controle, pensamento iterativo, uso da lógica condicional, depuração e detecção sistemática de erros.

Zanetti e Oliveira (2015) desenvolveram um estudo com objetivo de trazer uma proposta de ensino através da Robótica Pedagógica que possa amenizar as principais dificuldades de alunos iniciantes em programação, auxiliando nos aspectos relacionados a construção da solução de problemas. Os resultados demonstraram de maneira positiva que o método aplicado auxilia a composição da solução desenvolvida pelo aluno, permitindo que ele extraia informações mais concretas do objeto ou das ações sobre o objeto.

Oliveira (2007) apresenta sua experiência da utilização da robótica no ensino de matemática com alunos do 8º ano, concluindo que o interesse em solucionar os desafios propostos através da utilização dos robôs despertou a curiosidade e persistência dos alunos, provocando diversificadas estratégias, principalmente por tentativa e erro.

Ribeiro, Martins e Bernardini (2011) realizaram um estudo sobre o impacto da utilização da robótica em um ambiente educacional superior como ferramenta de apoio ao ensino de disciplinas de programação. Os autores constataram que a utilização da robótica no ensino de programação realmente apresenta um impacto positivo sobre os alunos, tornando-se um forte fator motivador e/ou estimulador sobre eles.

Tocháček, Lapeš e Fuglík (2016) desenvolveram um estudo com objetivo de avaliar o desenvolvimento do conhecimento tecnológico e habilidades de programação de alunos do ensino médio por meio de projetos de robótica educacional. Os resultados do estudo mostram que os projetos de robótica educacional representam uma importante ferramenta pedagógica, que pode ser usada como uma abordagem alternativa aos métodos tradicionais, aumentando a qualidade do processo educacional, a fim de desenvolver o conhecimento tecnológico e as habilidades de programação desses alunos.

Merkouris, Chorianopoulos e Kameas (2017) produziram um estudo com 36 estudantes de uma escola de ensino médio, a fim de explorar os benefícios que se obtém ao aprender a codificar computadores, como robôs e computadores vestíveis, em comparação com a programação do computador desktop. Descobriram que os alunos estavam mais envolvidos e tinham uma intenção maior de aprender a programar com o equipamento robótico do que com o computador de mesa.

Por sua vez, Gomes (2013) desenvolveu sua pesquisa com duas turmas do 10º ano na cidade de Funchal, na ilha da Madeira em Portugal, e afirma que a robótica é um método

motivador no processo de ensino-aprendizagem, promovendo o envolvimento e cooperação dos alunos para a construção do seu próprio conhecimento.

O presente estudo, diferente dos anteriores, possui um foco mais amplo, ao propor uma metodologia de ensino que reúne as técnicas da computação desplugada, programação visual e robótica educacional, com intenção de amenizar as dificuldades apresentadas por alunos iniciantes em programação do curso técnico em informática integrado ao ensino médio do IFNMG - *Campus* Montes Claros.

3 MATERIAIS E MÉTODOS

A pesquisa fundamenta-se na abordagem qualitativa e utiliza procedimentos técnicos da pesquisa de campo, buscando, através da opinião dos participantes, descrever os efeitos produzidos pela utilização de uma metodologia de ensino que faz uso de um *kit* de robótica educacional, desenvolvidos pelo autor deste trabalho, para o ensino de programação.

A pesquisa foi iniciada após aprovação pelo CEP¹, com previsão de encerramento em novembro de 2020. A submissão ao CEP visou a atender ao Art. 1º, Inciso VIII, § 2º da RESOLUÇÃO Nº 510, DE 07 DE ABRIL DE 2016:

Caso, durante o planejamento ou a execução da atividade de educação, ensino ou treinamento surja a intenção de incorporação dos resultados dessas atividades em um projeto de pesquisa, dever-se-á, de forma obrigatória, apresentar o protocolo de pesquisa ao sistema CEP/CONEP.(DIÁRIO OFICIAL DA UNIÃO, 2016).

A população estudada é composta por alunos da disciplina de Introdução à Programação, ofertada no primeiro ano do curso Técnico em Informática Integrado ao Ensino Médio do Instituto Federal do Norte de Minas Gerais – *Campus* Montes Claros. A instituição coparticipante ofereceu o espaço físico para aplicação do projeto. Como a população estudada é formada por adolescentes, o Termo de Consentimento Livre Esclarecido (TCLE) foi autorizado remotamente pelos pais ou responsáveis legais, por meio de gravação de áudio em virtude da pandemia da COVID-19. O termo de assentimento de menor (TAM) foi incorporado na primeira parte do formulário eletrônico, associado ao questionário da pesquisa, de forma que o participante pôde optar pela participação ou não na pesquisa. O contato dos alunos (*e-mail*) foi adquirido na sala de aula virtual da disciplina mencionada.

3.1 Ambiente de estudo

Os Institutos Federais são instituições de educação superior, básica e profissional, pluricurriculares e multicampi, especializados na oferta de educação profissional e tecnológica nas diferentes modalidades de ensino, com base na conjugação de conhecimentos técnicos e tecnológicos com as suas práticas pedagógicas.

O Instituto Federal de Educação, Ciência e Tecnologia do Norte de Minas Gerais (IFNMG) foi criado em 29 de dezembro de 2008, pela Lei nº 11.892, por meio da união do Centro Federal de Educação Tecnológica (CEFET) de Januária e da Escola Agrotécnica Federal de Salinas (EAF), instituições com mais de 50 anos de experiência na oferta da educação profissional. Atualmente possui 12 (doze) campi em funcionamento, na mesorregiões Norte e Noroeste de Minas e Vales do Jequitinhonha e Mucuri (IFNMG, 2019).

O IFNMG - *Campus* Montes Claros é um dos campi criado no processo de expansão da Rede de Educação Profissional e Tecnológica, localizado na região norte de Minas Gerais. A unidade foi implantada no ano de 2010, com funcionamento dentro instituições parceiras, até a construção da sede própria. Atualmente, o *Campus* atua em sede própria, localizada à Rua Dois,

¹ Parecer número CAAE: 36220220.0.0000.5108

300, Bairro Village do Lago I. O espaço escolar foi edificado em terreno doado pela Prefeitura à União e coube ao CEFET – MG a construção do prédio utilizando recursos repassados pelo Ministério da Educação do Governo Federal (MEC).

O *Campus* oferta cursos superiores e cursos para a formação técnica de nível médio voltados para o atendimento aos setores secundário e terciário da indústria. Nessa perspectiva, foi criado o Curso Técnico em Informática Integrado ao Ensino Médio, visando a atender os arranjos produtivos locais e os anseios da comunidade montesclarenses (IFNMG - CAMPUS MONTES CLAROS, 2020).

Os 30 estudantes que participaram deste experimento são do primeiro ano do curso Técnico em Informática Integrado ao Ensino Médio, com idade entre 13 e 15 anos. O projeto pedagógico do primeiro ano do curso é composto por 14 (quatorze) disciplinas do Núcleo Comum e 4 (quatro) disciplinas técnicas. As aulas ocorrem em dois turnos (matutino e vespertino), onde as disciplinas técnicas e propedêuticas intercalam (IFNMG - CAMPUS MONTES CLAROS, 2020). Normalmente, esses estudantes não têm experiência prévia em programação de computadores.

Dentre as diretrizes do curso Técnico em Informática, estabelecidas pelo MEC, encontra-se a criação de programas computacionais. A disciplina Introdução à Programação, do primeiro ano, é a base para o desenvolvimento de programas e envolve conteúdos relacionados a algoritmos e lógica matemática, porém é considerada a mais difícil por exigir um alto grau de abstração (BECKER, 2002; PEARS *et al.*, 2007; DENNING, 1989; JÚNIOR *et al.*, 2005; BENNEDSSEN; CASPERSEN, 2008; ZANETTI; OLIVEIRA, 2015).

Aprimorar a capacidade de abstração, decomposição e automação de processos é fundamental para resolução de problemas e obter êxito no ensino de programação. As próximas seções apresentam métodos que foram utilizados neste trabalho para desenvolver essas competências de maneira lúdica, e que conseqüentemente amplificam o pensamento computacional.

3.2 Metodologia de Ensino

A metodologia de ensino “RobôBlocks”, desenvolvida neste trabalho, foi aplicada durante as quatro primeiras semanas de aulas da disciplina de Introdução à Programação no ano de 2020, por meio de dois encontros semanais com duração de uma hora e quarenta minutos cada. A metodologia é dividida em três etapas, executadas de forma sequencial. Durante a primeira etapa, os alunos desenvolveram atividades desplugadas que estimulam habilidades envolvendo raciocínio lógico e resolução de problemas, sem o uso de computador. A atividade proposta está no Apêndice A, trabalha com conceitos de Matemática (Matrizes e Vetores) e Programação (Codificação, decodificação, desenvolvimento de raciocínio e lógica).

A segunda etapa requer uso de computador e conexão com a *internet*. Os alunos exercitaram a programação em blocos fazendo uso da plataforma Code.org. As atividades presentes na plataforma envolvem inúmeras habilidades. O objetivo foi que o aluno resolvesse as lições do curso 2, principalmente as que continham problemas relacionados a labirintos, quais

sejam: fase 3 (Labirinto: sequencia), fase 6 (Labirinto: laços), fase 8 (Abelha: laços), fase 10 (Abelha: depuração) e fase 13 (Abelha: condicional).

A terceira etapa foi desenvolvida no laboratório de informática da escola e precisou apenas dos computadores e do *kit* educacional “RobôBlocks”. Durante as atividades, os alunos utilizaram um computador individual, mas resolveram algumas tarefas em grupo. Nessa etapa, os alunos colocaram em prática os conceitos e habilidades adquiridas nas etapas anteriores, transformando o que era abstrato em real.

3.3 *Kit* Educacional Robôblocks

Até o momento, a plataforma Arduino não oferece oficialmente um *Kit* voltado para robótica educacional. Entretanto, são ofertados diversos *Kits* e componentes eletrônicos no mercado que podem ser utilizados com essa finalidade, contemplando do nível básico ao avançado.

A proposta do *Kit* Robôblocks é oferecer um robô que obedeça uma programação prévia, feita por blocos de códigos para seguir linha em um circuito (Figura 3), podendo ele ser automatizado com movimentos planejados. Com a utilização do *Kit*, é possível desenvolver atividades com foco na resolução de problemas associados a labirintos, instigando os alunos a serem criativos, tomarem decisões, fazerem reflexões e darem significado aos conteúdos apresentados em sala de aula.

Figura 3 – Robô seguindo linha



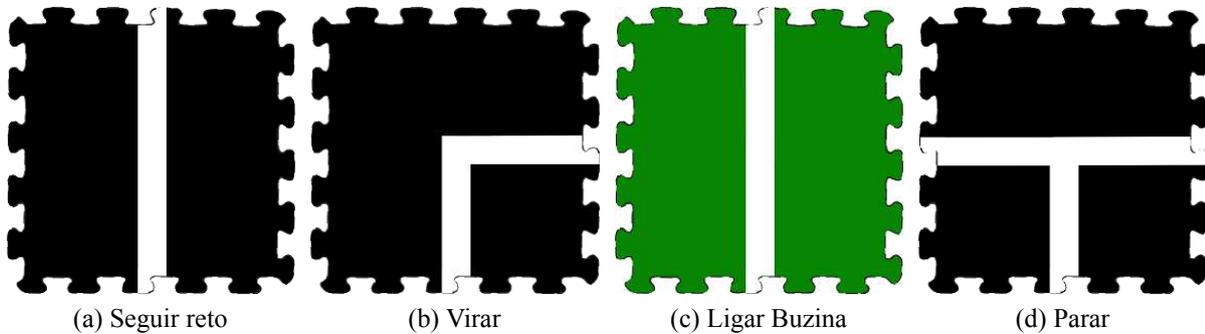
Fonte: Próprio autor

O equipamento robótico foi montado pelo professor da disciplina. Destaca-se que a montagem desse equipamento robótico é simples e qualquer pessoa, mesmo sem conhecimento de robótica, pode montá-lo utilizando apenas o manual de instruções, presente no Anexo A, para realizar a conexão das peças mencionadas na seção 3.3.2.

O *Kit* contém 22 peças de tapete para tatame com medidas de 50cm x 50cm x 1cm, confeccionadas em EVA preto e cortadas em formato de quebra cabeça. Para montar o circuito estão disponíveis: 11 peças com uma faixa branca reta (Figura 4a); 08 peças com uma faixa branca em forma de curva (Figura 4b); 02 peças verdes com uma faixa branca reta usada para

acionar a buzina (Figura 4c) e 01 Peça para fim de percurso (Figura 4d). A faixa branca é feita com fita de demarcação de solos com 5cm de largura.

Figura 4 – Peças do Tabuleiro

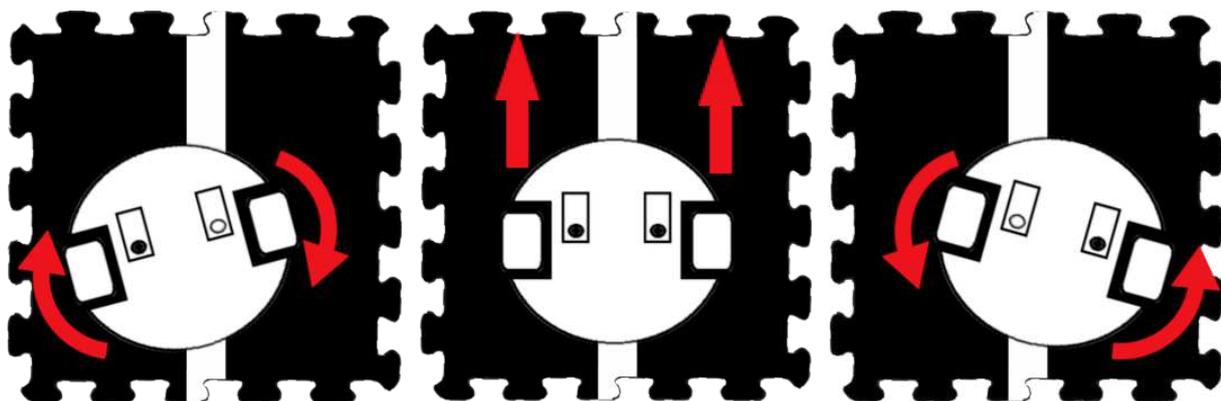


Fonte: Próprio autor

3.3.1 Como o robô (carrinho) seguidor de linha funciona?

O robô seguidor de linha funciona de maneira bem simples. Ele deve seguir andando por cima de uma linha de cor preta ou branca. Nesse caso, deve-se assumir que a pista é preta e a faixa é branca. O robô possui 2 sensores infravermelho que detectam a presença ou não dessa faixa. De acordo com a combinação dos sensores, o carrinho irá para frente ou virará para um dos lados, conforme Figura 5.

Figura 5 – Movimentos do robô para seguir a linha.

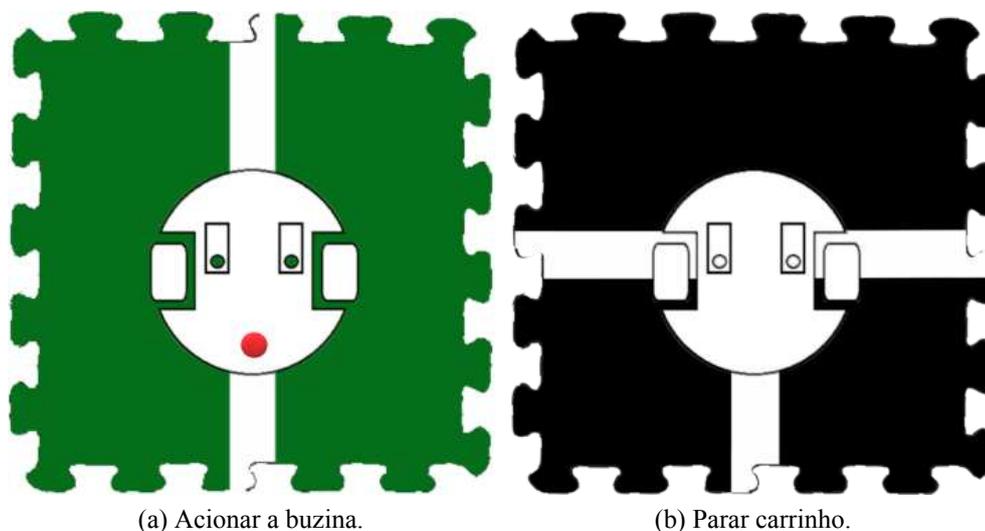


- (a) Carrinho deve virar para direita: sensor da direita em nível baixo (detectou a linha), o motor da esquerda deve rodar mais rápido agora.
- (b) Carrinho seguindo para frente.
- (c) Carrinho deve virar para esquerda: sensor da esquerda em nível baixo (detectou a linha), o motor da direita deve rodar mais rápido agora.

Fonte: Próprio autor

Além dos movimentos descritos na Figura 5, o robô deve reconhecer o piso verde (Figura 6a) e o fim do circuito (Figura 6b). No piso verde o robô deve acender um LED ou emitir um sinal sonoro, a quantidade de vezes que for necessário. No fim do circuito, o robô deve parar desligando os motores, ao reconhecer presença de linha branca em ambos os sensores.

Figura 6 – Reconhecimento extra

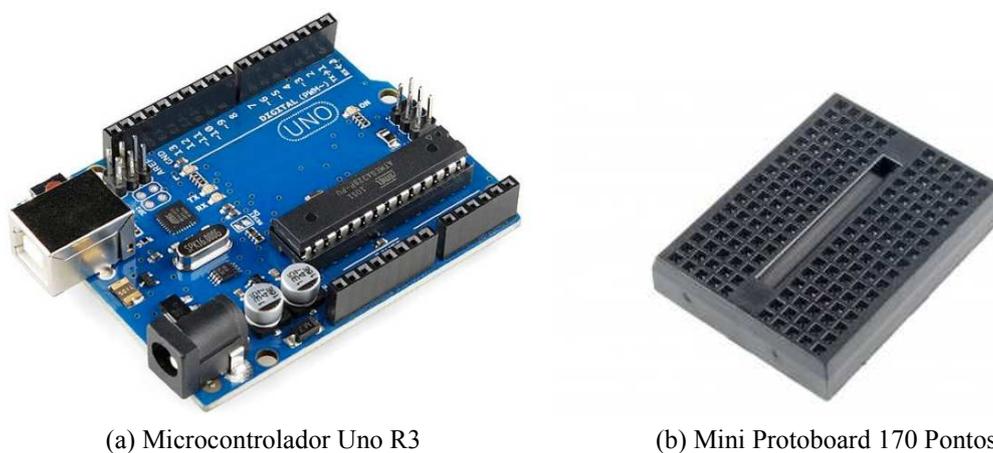


Fonte: Próprio autor

3.3.2 Componentes e Montagem do Kit

A placa de prototipagem designada para o projeto foi o arduino UNO (Figura 7a), por ser robusta, barata e possuir uma quantidade de portas suficiente para conectar os sensores e atuadores do robô. Para ampliar a pinagem do arduino, utilizou-se uma mini protoboard de 170 pontos (Figura 7b), por ser leve e discreta. A buzina (*buzzer*) deve ser conectada diretamente à protoboard, seguindo a indicação de porta do manual de instruções (Anexo A).

Figura 7 – Placa de prototipagem



(a) Microcontrolador Uno R3

(b) Mini Protoboard 170 Pontos

Fonte: FILIPEFLOP, 2019.

O chassi aplicado a esse projeto é feito em acrílico no formato redondo (Figura 8a) e utiliza dois motores DC (*Direct Current*) com redução (Figura 8b) e uma roda boba (Figura 8c), facilitando assim o desempenho do robô no tabuleiro. Essas partes devem ser montadas

primeiro para que possam receber os demais componentes, como a placa Arduino e a protoboard que ficam posicionadas na parte superior do chassi.

Figura 8 – Kit Chassi Redondo 2WD Robô

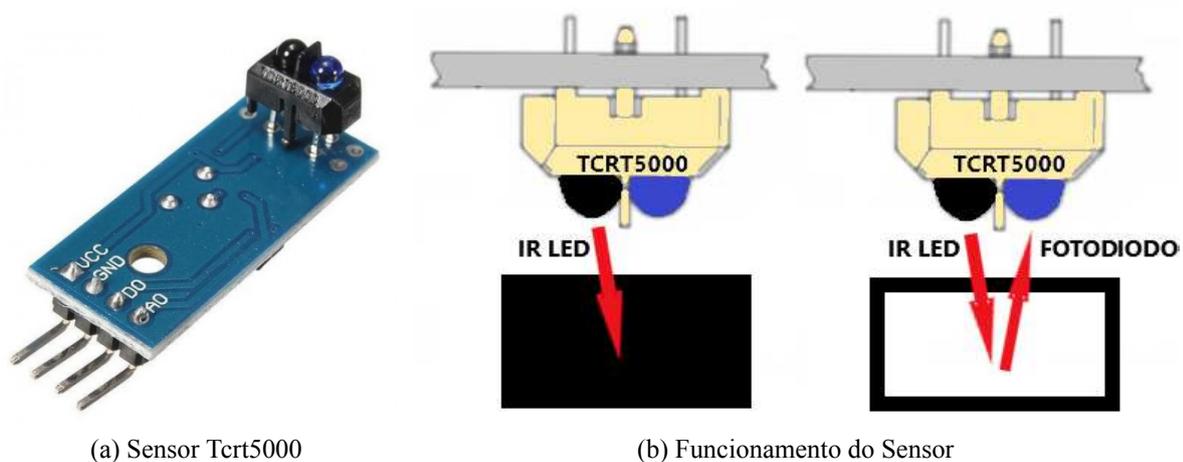


Fonte: FILIPEFLOP, 2019.

3.3.2.1 Sensor de linha infravermelho Tcrt5000

O módulo TCRT5000 (Figura 9a) possui um sensor de obstáculo infravermelho IR composto basicamente por um LED emissor e um fototransistor responsável por detectar a luz infravermelha. Ele emite continuamente uma luz infravermelha por um LED negro e capta o reflexo com um LED claro. Portanto, como a luz reflete em superfícies claras e é absorvida em superfícies negras, conforme é ilustrado na Figura 9b, o LED receptor irá detectar a luz infravermelha no branco e não detectar no preto (VIDA DE SILÍCIO, 2019). O sensor deve ser posicionado na parte inferior do chassi, de maneira que fique próximo do solo onde irá fazer a leitura da refletância.

Figura 9 – Sensor de linha infravermelho



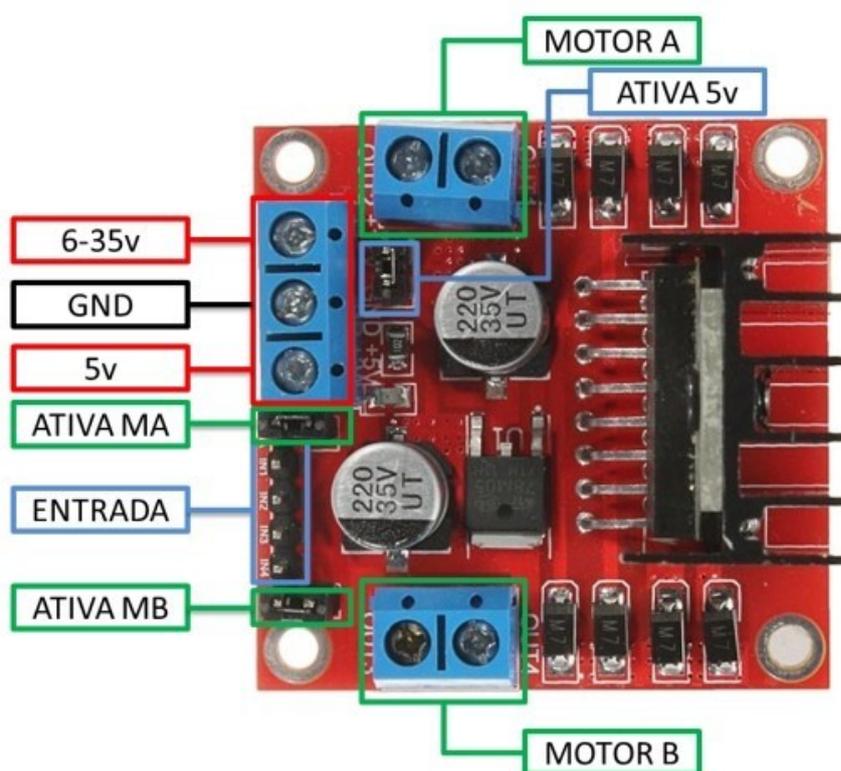
Fonte: FILIPEFLOP, 2019.

3.3.2.2 Ponte H L298

Para controlar os motores DC a partir de sinais produzidos pelo microcontrolador, utiliza-se um módulo conhecido como “Ponte H”. Esse módulo possui um circuito integrado, e é utilizado para controlar atuadores com cargas indutivas. Permite o controle não só do sentido de rotação do motor, bem como a sua velocidade, quando ligado nos pinos PWM do Arduino. O módulo Ponte H é amplamente utilizado em projetos de robótica, principalmente por possuírem dimensões pequenas, o que facilita sua acomodação no robô (VIDA DE SILÍCIO, 2019).

É possível construir a ponte H utilizando-se relés, transistores ou chaves. Entretanto, foi utilizado neste projeto o módulo L298N (Figura 10) com o intuito de simplificar a montagem do Robô. Esse controlador deve ser posicionado na parte inferior do chassi, facilitando sua conexão com os motores.

Figura 10 – Módulo Ponte H L298N

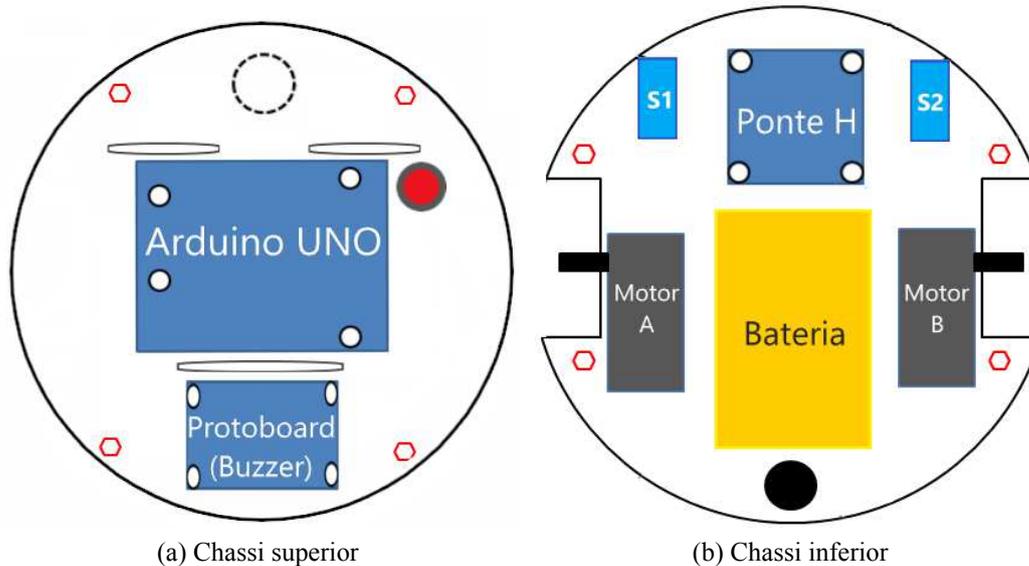


(a) Entradas e saídas do módulo Ponte H L298n

Fonte: VIDA DE SILÍCIO, 2019

Na Figura 11 é mostrada a disposição dos componentes no chassi superior e inferior, o que é de suma importância, pois o Arduino detecta o ambiente recebendo entradas de sensores e afeta seus arredores controlando luzes, motores e outros atuadores (ARDUINO, 2019).

Figura 11 – Disposição dos componentes no chassi



(a) Chassi superior

(b) Chassi inferior

Fonte: Próprio autor

Após a montagem do robô, deve-se ficar atento com a fiação que interliga os componentes (Anexo A), para que esteja em conformidade com a programação enviada para o microcontrolador da placa Arduino, e o conjunto de instruções execute de forma correta. A proposta do equipamento está em consonância com o *software* DuinoBlocks4Kids, que foi adaptado para atuar com os componentes mencionados na seção 3.3.2, não podendo ser aplicado a outro *Kit* qualquer, a não ser que tenha os mesmos sensores e atuadores conectados às mesmas entradas.

3.4 Software

Com o carrinho montado, deve-se utilizar o DuinoBlocks4Kids adaptado para programá-lo. As instruções de instalação, configuração e uso estão no Anexo A. As adaptações do DB4K foram feitas nos blocos que serão descritos a seguir, para melhor atender os objetivos do *Kit*:

- Bloco do Sensor de refletância (Figura 12): foi incluído a opção de trabalhar com um sensor de refletância central, quando selecionado “Centro”, conectado ao pino “A4” para melhor leitura do percurso. Também foi inserida nesse bloco a opção “Media”, para reconhecer a cor verde que fica entre a branca/baixa e a preta/alta, de acordo com a refletância da superfície.

Figura 12 – Bloco do sensor de refletância



Fonte: Próprio autor.

- Bloco da Buzina (Figura 13): modificado para permitir ligar a buzina quando selecionado “Sim” e desligar a buzina quando selecionado “Não”. Conectado por meio do pino 7.

Figura 13 – Bloco da buzina



Fonte: Próprio autor.

Depois da estruturação do conjunto de blocos necessários para percorrer o circuito, é necessário apertar o botão laranja presente na parte superior da área de trabalho dos blocos. Esse botão envia o programa diretamente para a placa Arduino. Caso, por alguma razão, a plataforma do DB4K não consiga enviar o programa para a placa, pode-se utilizar a opção de Menu/Código Arduino/Abrir código na IDE Arduino ou copiar o código gerado e colar no Arduino IDE (Ambiente Integrado de Desenvolvimento).

3.5 Testes Realizados com o *Kit RobôBlocks*

Foram realizados diversos testes com o protótipo, buscando analisar a performance dos sensores, atuadores e o desempenho do *Kit* como um todo. O experimento foi feito no tabuleiro com 20 peças de tatame em EVA de 50cm x 50cm x 1cm encaixadas para montar o circuito. O Robô percorreu o circuito (seguindo linha) obedecendo aos comandos previamente programados na plataforma DB4K adaptada.

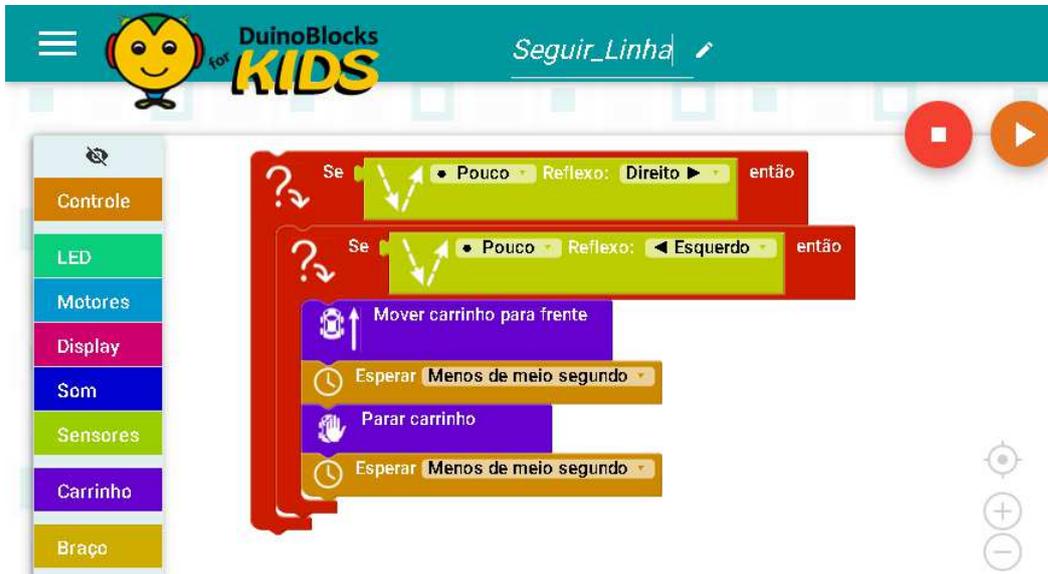
Após serem testadas variadas sequências de blocos de código, as que melhor atenderam os objetivos do *Kit* estão representadas nas Figuras 14 a 18:

Figura 14 – Avançar com o buzzer ligado quando reconhecer o verde nos sensores da direita e esquerda



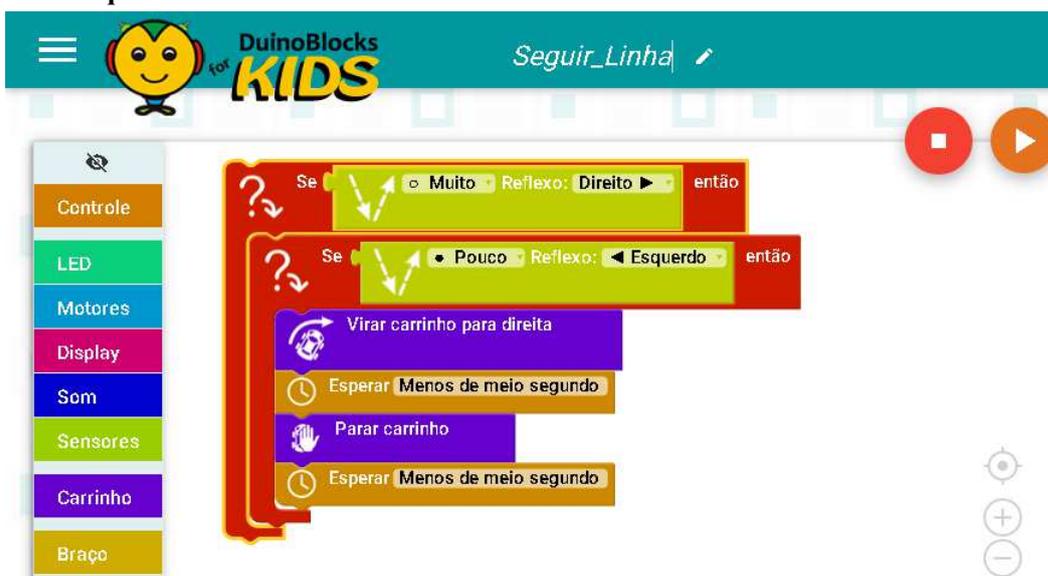
Fonte: Próprio autor

Figura 15 – Avançar quando reconhecer o preto nos sensores da direita e da esquerda



Fonte: Próprio autor

Figura 16 – Virar a direita quando reconhecer o branco no sensor da direita e o preto no sensor da esquerda



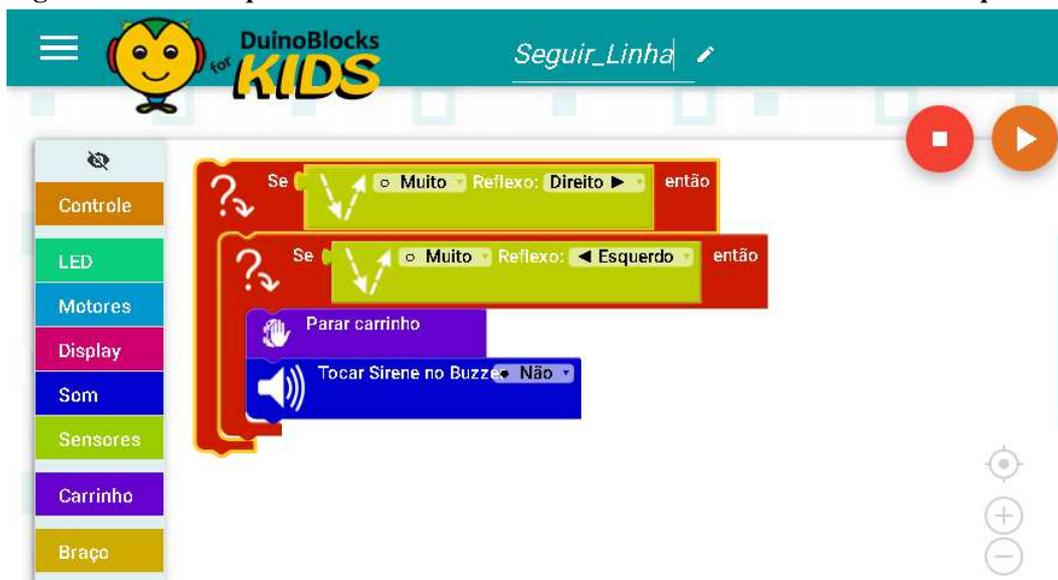
Fonte: Próprio autor

Figura 17 – Virar a esquerda quando reconhecer o branco no sensor da esquerda e o preto no sensor da direita



Fonte: Próprio autor

Figura 18 – Parar quando reconhecer o branco nos sensores da direita e da esquerda



Fonte: Próprio autor

3.6 Custos para montagem do *Kit* Robôblocks

De acordo com Filho e Gonçalves (2008), muitas pesquisas indicam a robótica educacional como sendo um instrumento facilitador do processo ensino-aprendizagem, que envolve assuntos multidisciplinares. Entretanto, os autores enfatizam que ela não faz parte do cotidiano das escolas brasileiras, pela dificuldade na aquisição do equipamento que, basicamente, ainda possui um custo impeditivo.

Diante dessa conjuntura e entendendo a relevância que um equipamento dessa natureza possui para o desenvolvimento de um processo educacional moderno e dentro da realidade das escolas brasileiras, é que surgiu o estímulo para o desenvolvimento de um *Kit* educacional com funcionalidades semelhantes às existentes em conjuntos de robótica educacional disponíveis no mercado, porém, com custo reduzido. Um exemplo desses conjuntos temos o *Kit LEGO mindstorms* que custa em média R\$ 5.000,00 (EXTRA, 2020).

A construção do *Kit* transcorreu a partir de diversos componentes: placa de prototipagem (Arduino Uno R3), atuadores (Motores DC com Redução, botão liga/desliga, buzina, ponte H), sensores (sensor de linha infravermelho), dentre outros, resultando em um projeto de custos acessíveis (Tabela 1).

Tabela 1 – Custos para montagem do *Kit*

Componentes	Quantidade	Custo
Arduino Uno R3 + Cabo USB	1	R\$ 39,80
Chassis em acrílico	2	
Motores DC com Redução	2	
Rodas com Pneus	2	
Roda Boba	1	
Espaçadores para Roda Boba	12	
Espaçadores para Placa Arduino UNO	4	
Alongadores 37mm (Amarelo)	4	R\$ 75,90
Porcas M3	8	
Parafusos Soberba de 11mm para Alongadores	8	
Parafusos Soberba de 11mm para Placa UNO	4	
Parafusos M3 18mm	4	
Parafusos M3 30mm	4	
Suporte de acrílico para fixar o motor da roda	4	
Botão liga/desliga	1	R\$ 04,00
Protoboard SYB-170 Mini	1	R\$ 09,99
Sensor De Linha Infravermelho Ir Tcrt5000 Lm393	2	R\$ 12,45
Buzina	1	R\$ 02,00
Ponte H L298 H-bridge Motor Driver Board 2a L298n	1	R\$ 24,90
Jumpers- Macho/Macho	1	R\$ 09,80
Baterias Recarregável 9v 450 Mah	2	R\$ 55,90
Carregador de bateria 9v	1	
<i>Kit</i> 20 Tapete Eva 50x50x1 Cm	25	R\$120,00
Fita Para Demarcação De Solo - branca - 48mm x 30m	1	R\$ 35,00
Total		R\$ 389,74

3.7 Coletas de dados

Dentre os 30 estudantes que utilizaram a metodologia de ensino durante as aulas da disciplina de Introdução à Programação, ofertada no primeiro ano do curso Técnico em Informática Integrado ao Ensino Médio do IFNMG - Campus Montes Claros, em 2020, foi aplicado um questionário com uma pesquisa de opinião a 18 estudantes voluntários, perante aprovação do CEP e autorização dos responsáveis. Dado que o autor deste trabalho é o professor da disciplina, utilizou-se também a sua observação durante a execução da metodologia para corroborar os resultados e discussões.

O questionário (Apêndice B) foi aplicado após o experimento, e fez uso de questões abertas e fechadas para avaliar, na percepção do aluno, se a metodologia de ensino e o *Kit* educacional RobôBlocks estimularam habilidades e competências necessárias para o desenvolvendo do pensamento computacional e amenizaram as dificuldades na aprendizagem do curso. A aplicação do questionário ocorreu por meio de formulário eletrônico e as respostas não identificam o participante, de forma a garantir a privacidade e confidencialidade das informações.

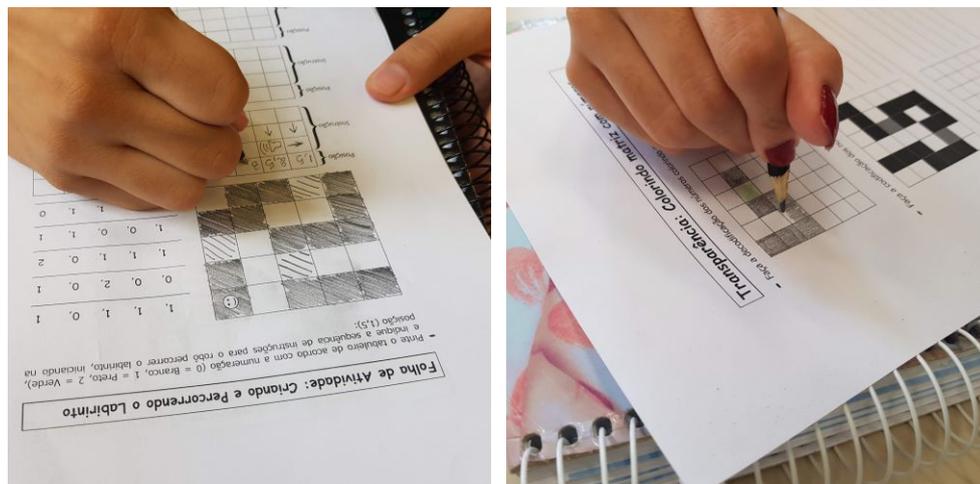
No que se diz respeito aos riscos para os participantes, alguns cuidados foram tomados a fim de evitar que eles se sentissem constrangidos ao responderem o questionário. O participante poderia se sentir constrangido em responder o formulário se entendesse que isso influenciaria de alguma forma em sua avaliação na disciplina. Para evitar isso, foi feita uma conversa prévia por meio do contato de *WhatsApp* disponível no grupo da disciplina, para que o participante se sentisse à vontade em responder as questões e, caso ele continuasse constrangido ou desconfortável com alguma questão, poderia optar pela alternativa “prefiro não responder” ou até mesmo desistir de responder ao questionário. Além disso, o pesquisador deixou claro aos participantes que não haveria penalização em virtude do não preenchimento do questionário.

4 RESULTADOS E DISCUSSÕES

Esta pesquisa busca avaliar se uma metodologia de ensino que utiliza um *Kit* de robótica educacional de baixo custo, ameniza as dificuldades apresentadas por alunos iniciantes em programação. A metodologia de ensino, denominada “RobôBlocks”, foi aplicada como projeto piloto durante as primeiras semanas de aula da disciplina de Introdução à Programação, por meio do professor da disciplina que é o autor deste trabalho. A turma selecionada para realizar a pesquisa, foi a do 1º ano do curso técnico em informática integrado ao ensino médio do Instituto Federal - *Campus* Montes Claros no ano de 2020. Todos os 30 alunos dessa turma participaram das atividades propostas, no entanto, somente 18 foram autorizados a participar da pesquisa.

Na primeira semana, nos dias 05 e 06 de fevereiro desse mesmo ano, foi realizada a prática com a computação desplugada, momento em que os estudantes desenvolveram atividades sem o uso do computador, conforme apresentado na Figura 19. Os estudantes exercitaram conceitos de Matemática (Matrizes e Vetores) e Programação (Codificação, decodificação, desenvolvimento de raciocínio e lógica), dando e seguindo instruções para solucionar problemas.

Figura 19 – Atividade 1 - Computação Desplugada



Fonte: Próprio autor

Durante a atividade desplugada o professor explicou conceitos básicos necessários para desenvolver um algoritmo como, por exemplo: o que é uma instrução; qual a necessidade de criar uma sequência lógica; como funciona um vetor e uma matriz.

Nessa etapa, foi possível perceber o entusiasmo dos participantes em solucionar os problemas relacionados aos labirintos. Ao executar o passo a passo para percorrer o labirinto, os alunos conseguiam identificar e corrigir os erros de lógica. Os alunos utilizaram a criatividade e uma estratégia particular para realizar os exercícios propostos e fizeram comparações com os colegas para avaliar a melhor solução.

Na segunda semana, nos dias 12 e 13 de fevereiro, foram desenvolvidas atividades que envolveram a programação em blocos, por meio do curso 2 da plataforma CODE.org, onde

os estudantes utilizaram blocos de código em componentes visuais (Figura 20). Essa atividade permitiu que, de forma lúdica, os estudantes focassem apenas na lógica de funcionamento dos blocos, sem se preocupar com a complexa sintaxe de uma linguagem de programação.

Figura 20 – Atividade 2 - Programação em Blocos no CODE.org

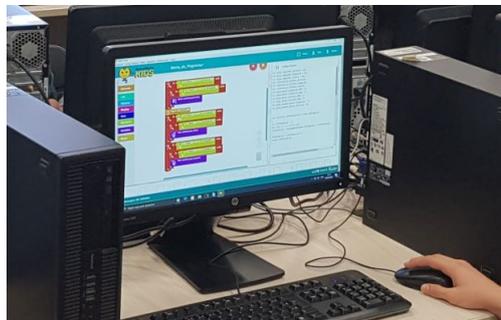


Fonte: Próprio autor

A princípio, alguns estudantes tiveram uma certa dificuldade em se localizar dentro do tabuleiro para realizar o percurso necessário. No entanto, percebeu-se que essa dificuldade foi diminuindo no decorrer das atividades do curso. Notou-se também que os envolvidos na execução das atividades aprenderam, por meio da prática, conceitos básicos como sequência lógica, laços de repetição, depuração de erros e condicionais.

Na terceira semana, nos dias 19 e 20 de fevereiro, os estudantes colocaram em prática as habilidades e competências adquiridas nas semanas anteriores, transformando o que era abstrato em real, por meio da robótica. Durante as atividades, os alunos utilizaram um computador individual, mas resolveram algumas tarefas em grupo quando não conseguiam sozinhos, tal qual, por exemplo, a programação para o equipamento seguir a linha (Figura 21).

Figura 21 – Atividade 3 - Programação em Blocos para Arduino no DB4K



Fonte: Próprio autor

Nesse momento, percebeu-se a motivação dos estudantes em programar o equipamento e fazer com que ele executasse os comandos necessários para automatizar seus procedimentos. Novamente, de maneira lúdica, os estudantes estimularam suas habilidades e competências fundamentais para analisar e desenvolver um algoritmo. Dentre elas, temos principalmente:

dar e seguir instruções; abstração e decomposição de problemas; aprendizado através dos erros; automação de procedimentos; raciocínio lógico e resolução de problemas.

Já que foi utilizado somente um equipamento robótico, os estudantes produziam a programação com a montagem da sequência de blocos e, somente quando terminavam, solicitavam o equipamento para testar o código. O professor da disciplina percebeu a necessidade de mais equipamentos para facilitar os testes e melhorar a dinâmica das aulas e, além disso, acrescentar no mínimo 3 encontros de 1 hora e 40 minutos para desenvolver as atividades.

Para finalizar a aplicação da metodologia, no dia 27 de fevereiro, os estudantes dividiram-se em equipes, utilizaram a criatividade para montar o tabuleiro com o percurso que o equipamento deveria seguir (Figura 22), e desenvolveram a programação do equipamento por meio do DB4K adaptado, a fim de automatizar seus procedimentos para seguir linha, acionar a buzina quando encontrar o piso verde e parar no final do percurso. A sequência de blocos de código foi enviada para o equipamento para ser testada no percurso do tabuleiro (Figura 23). Nesse momento, os estudantes, por meio da tentativa e erro, puderam verificar qual parte da programação necessitava de correção.

Figura 22 – Atividade 3 - Montagem do percurso



Fonte: Próprio autor

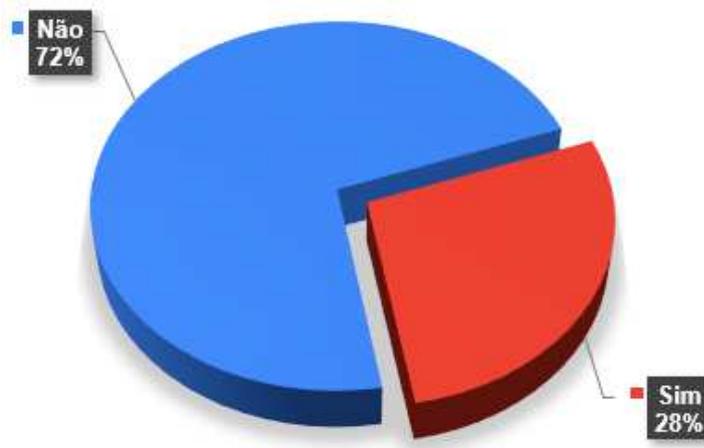
Figura 23 – Atividade 3 - Testando a Programação



Fonte: Próprio autor

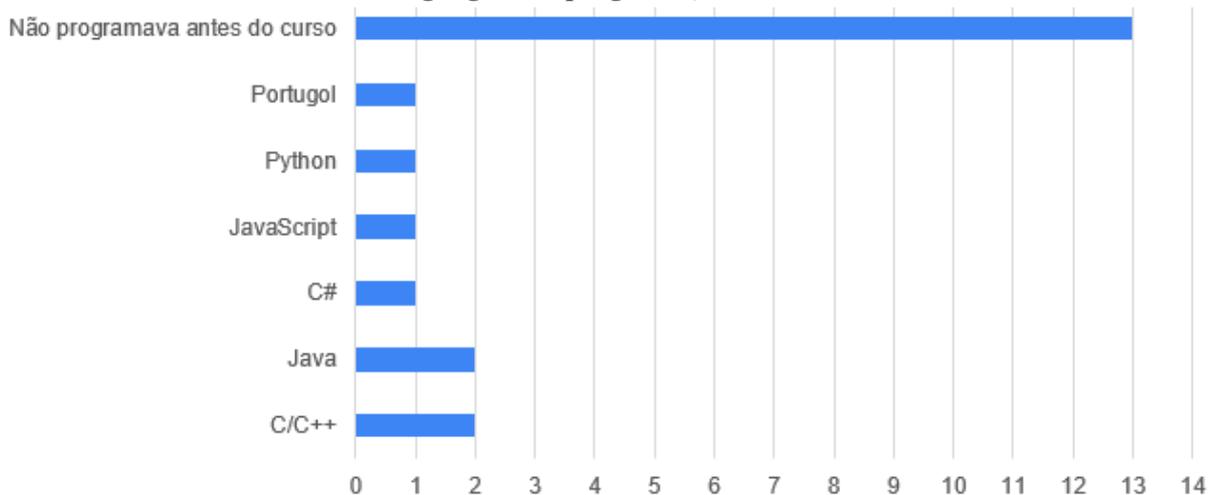
Visto que, durante a execução da metodologia de ensino, surgiu a intenção de investigar e incorporar os resultados a este trabalho, foi realizada uma pesquisa de opinião com os participantes da disciplina. O questionário foi respondido por 18 estudantes dos 30 que fizeram uso da metodologia de ensino ‘RobôBlocks’. Desses estudantes, somente 5 tiveram contato com alguma linguagem de programação antes de entrar no curso, representando 28% conforme ilustrado no Gráfico 1. Dentre as linguagens utilizadas pelos estudantes antes de entrar no curso, C/C++ e Java foram as mais citadas, contemplando 50% entre as demais, conforme ilustrado no Gráfico 2, sendo que todas tinham finalidade de estudo ou curiosidade.

Gráfico 1 – Você teve contato com alguma linguagem de programação antes de entrar no curso?



Fonte: Próprio autor.

Gráfico 2 – Quais linguagens de programação você costumava utilizar?

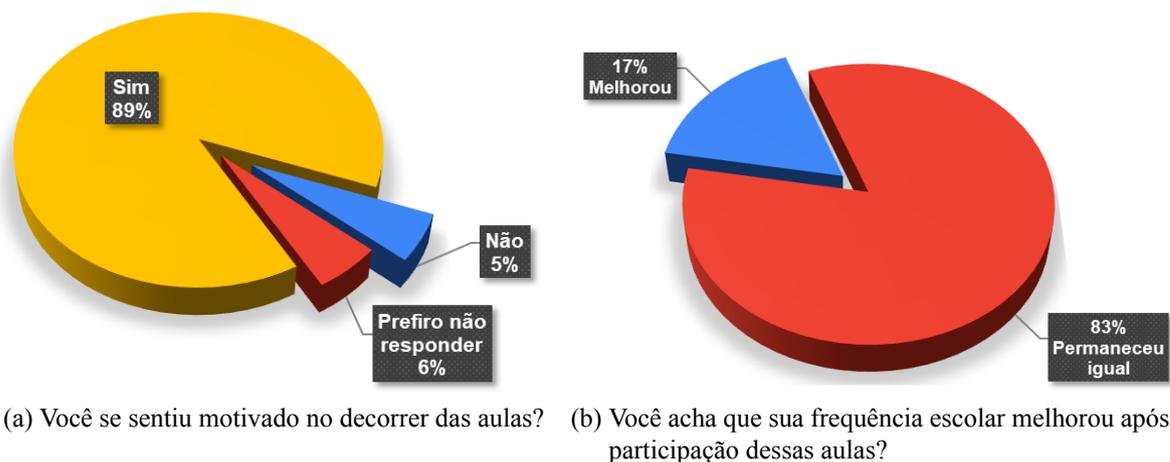


Fonte: Próprio autor.

Os resultados observados são promissores, principalmente considerando que 89% dos estudantes respondentes se sentiram motivados no decorrer das aulas com a metodologia de ensino RobôBlocks, conforme ilustrado no Gráfico 3a. A motivação pode ter contribuído

positivamente na frequência dos estudantes durante as aulas do curso que, de acordo com o ponto de vista de 17% dos respondentes, obteve uma melhora após a participação dessas aulas (Gráfico 3b).

Gráfico 3 – Metodologia de ensino RobôBlocks



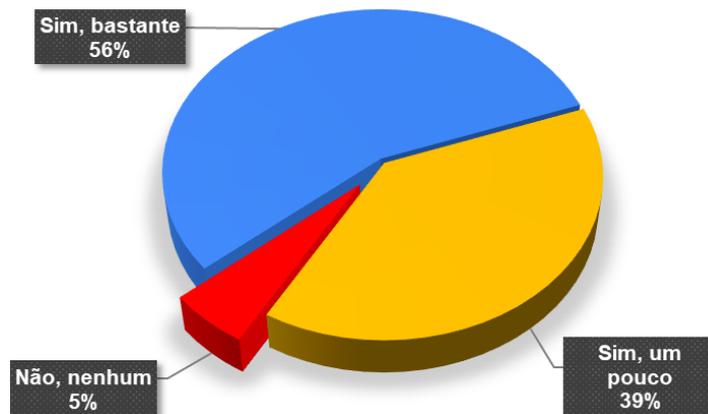
Fonte: Próprio autor

Conforme declaração de 14 estudantes, o desempenho escolar no curso teve uma melhora significativa, que pode estar associada a mudança na forma de pensar, bem como a motivação provocada pela metodologia. Benitti (2012) expõe que entre os métodos e ferramentas de ensino possíveis, destaca-se a Robótica Pedagógica, que quando presente durante as aulas pode trazer experiências educacionais mais completas e um retorno mais rápido de habilidades e competências aprendidas pelos estudantes.

Das 4 estudantes do sexo feminino que participaram desta pesquisa, mesmo sem ter tido contato com uma linguagem de programação antes do curso, todas sentiram-se motivadas e acharam que o seu desempenho escolar na disciplina melhorou mediante a participação das aulas do projeto de ensino RobôBlocks. Em contra-partida, dos 14 estudantes do sexo masculino que participaram desta pesquisa, 5 já tiveram contato com uma linguagem de programação antes do curso, 12 sentiram-se motivados e somente 6 acharam que o seu desempenho escolar na disciplina melhorou mediante a participação das aulas do projeto de ensino RobôBlocks. Nesse contexto, Rusk *et al.* (2008) apontam que a robótica educacional pode produzir uma interdisciplinaridade relevante, explorando uma série de possíveis aplicações para envolver os estudantes com diferentes interesses.

Percebe-se que os alunos gostaram da metodologia e consideraram o conteúdo trabalhado importante para o seu desempenho na disciplina de Introdução à Programação. No Gráfico 4 é mostrado a porcentagem de alunos que afirmaram ter adquirido conhecimentos importantes para desenvolver um programa. Dentre os participantes da pesquisa, 94,4% ficaram satisfeitos com essas aulas e 100% deles recomendaria essas aulas para outro colega.

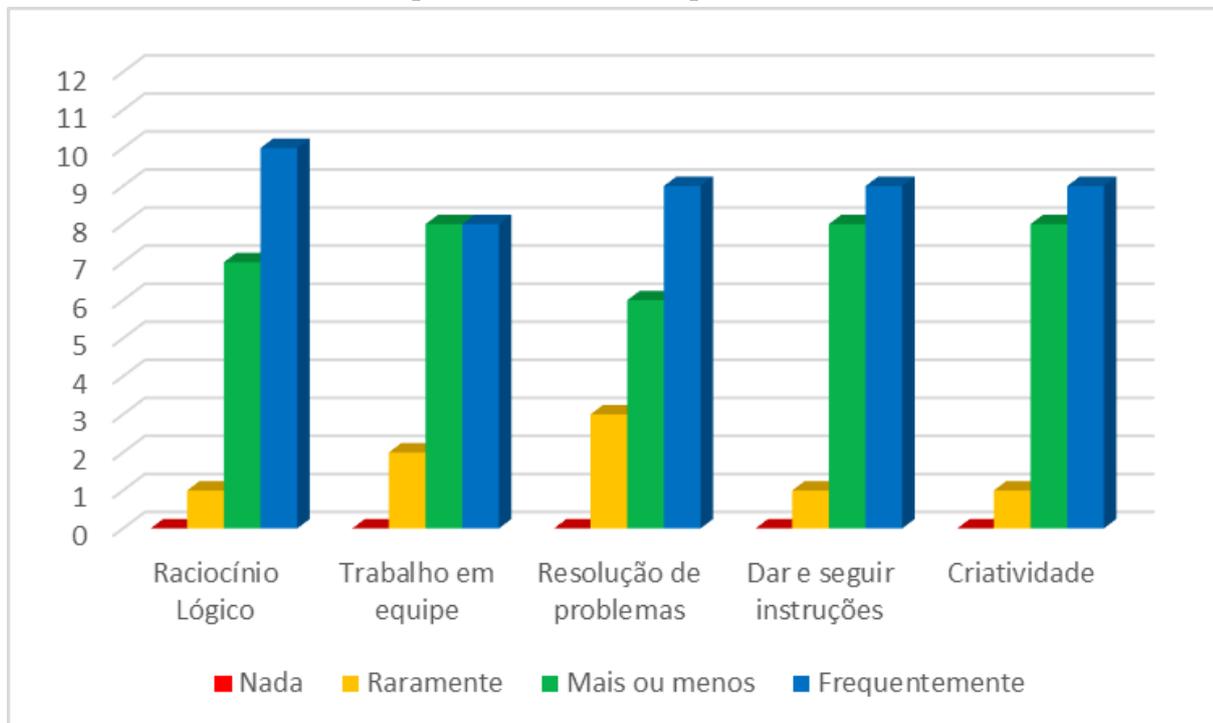
Gráfico 4 – Você acredita que obteve conhecimentos importantes para desenvolver um programa?



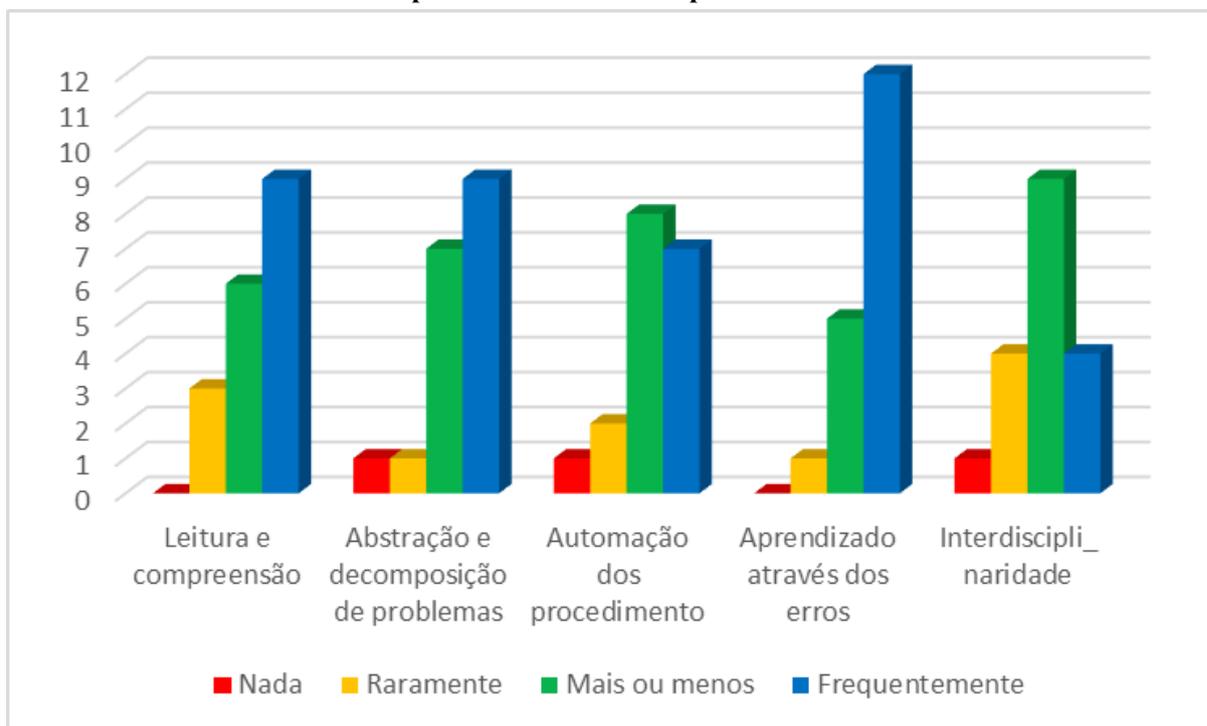
Fonte: Próprio autor.

Por meio do questionário os estudantes puderam apontar em uma escala de 1 a 4 (sendo que 1 corresponde a “Nada”, 2 a “Raramente”, 3 a “Mais ou Menos” e 4 a “Frequentemente”), de acordo com sua opinião, se as habilidades e competências apresentadas no Gráfico 5 e 6 foram estimuladas por meio da utilização do *Kit* Educacional RobôBlocks. Faz-se necessário ressaltar que os conceitos destas habilidades e competências foram trabalhados no decorrer das aulas do projeto, na medida em que eram aplicadas.

Gráfico 5 – Habilidades e Competências estimuladas por meio do *Kit* Educacional RobôBlocks

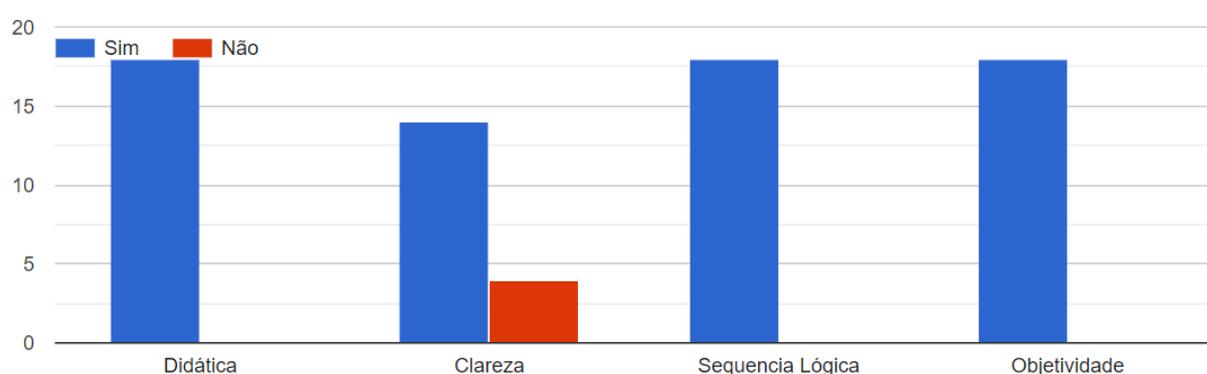


Fonte: Próprio autor.

Gráfico 6 – Habilidades e Competências estimuladas por meio do *Kit* Educacional RobôBlocks

Fonte: Próprio autor.

É notório que, do ponto de vista dos estudantes, a grande maioria das habilidades e competências foram estimuladas durante as aulas, sendo que somente três, dentre as dez questionadas, tiveram voto negativo, o que não impactou na avaliação da metodologia. Analisando o Gráfico 7, pode-se afirmar que os estudantes consideraram as características da metodologia satisfatórias, pois embora o quesito clareza tenha tido votos negativos, atingiu 77,8% positivos.

Gráfico 7 – A metodologia de ensino proposta para aplicação do *kit* possui as seguintes características?

Fonte: Próprio autor.

Em suma, vale mencionar os cinco comentários tecidos pelos alunos participantes da pesquisa, com o objetivo de melhor ilustrar os resultados obtidos por meio deste estudo:

“Uma maneira fácil e descomplicada de aprender programação. A metodologia de ensino RobôBlocks não é estressante e massiva, e sim descontraída e divertida.”

“Achei a metodologia muito interessante e acredito ter aprendido bastante.”

“Gostei muito da metodologia.”

“Essa metodologia é um bom modo de acolher os alunos novatos na programação, pois é bem simples e fácil de aprender o básico. Eu aprovei o método e acho que deveria ser usado frequentemente.”

“É ideal para introduzir a lógica para possíveis novos programadores, faz com que essa tarefa se torne intuitiva e que os alunos se ajudem entre si, para quem nunca teve uma experiência na área é um início simples e tranquilo, que deixará uma base para avançar no conteúdo.”

Por meio dos comentários, pode-se perceber que a metodologia de ensino RobôBlocks teve uma boa aceitação, deixando mais simples e agradável o primeiro contato com a disciplina de Introdução à Programação, o que tende a diminuir a retenção e evasão escolar.

Para melhor aproveitar a metodologia sugerida, recomenda-se a aplicação por meio de oficina, de modo paralelo às aulas da disciplina, com ampliação do tempo de execução de 7 para 10 encontros de 1 hora e 40 minutos. A coleta de dados para análise da metodologia pode ser melhorada com a aplicação de uma avaliação diagnóstica antes e uma depois da oficina, a fim de comparar a evolução das habilidades e competências dos envolvidos (Gráfico 8).

Gráfico 8 – Distribuição das atividades da oficina sugerida

Execução	1ª Aula	2ª Aula	3ª Aula	4ª Aula	5ª Aula	6ª Aula	7ª Aula	8ª Aula	9ª Aula	10ª Aula
1ª Avaliação diagnóstica	■									
1ª Etapa - Atividade Desplugada	■	■	■							
2ª Etapa - Programação em Blocos				■	■	■				
3ª Etapa - Robótica Educacional							■	■	■	■
2ª Avaliação diagnóstica										■

Fonte: Próprio autor.

5 CONSIDERAÇÕES FINAIS

Este trabalho mostrou a construção do *Kit* Educacional RobôBlocks e uma metodologia de ensino para sua aplicação. A proposta foi desenvolvida para apoiar o ensino de programação, no entanto pode ser aplicada a estudantes de qualquer área do ensino, já que foi desenvolvida para que os estudantes sejam instigados a serem criativos, tomarem decisões, fazerem reflexões e darem significado aos conteúdos apresentados em sala de aula.

Aplicou-se a metodologia durante as aulas da disciplina de introdução à programação, porém, na opinião do professor, seria melhor aplicar em paralelo às aulas por meio de oficina, ampliando o tempo para execução das atividades. O professor recomenda 10 encontros de 1 hora e 40 minutos, sendo 3 encontros para primeira etapa, 3 encontros para segunda etapa e 4 encontros para terceira etapa. O professor também percebeu a necessidade de mais equipamentos para facilitar os testes e melhorar a dinâmica das aulas.

A avaliação da metodologia de ensino foi realizada a partir dos dados coletados por meio da observação do professor e do questionário aplicado aos estudantes, visando a investigar o quanto o método proposto promove o desenvolvimento do raciocínio lógico e auxilia no ensino de programação. Embasando-se nos resultados obtidos, pode-se dizer que o objetivo do trabalho, ainda que com poucos participantes, foi alcançado, visto que a metodologia de ensino “RobôBlocks”, que faz uso de um *Kit* de robótica educacional de baixo custo, mostrou-se eficiente em amenizar as dificuldades apresentadas por alunos iniciantes em programação.

Por fim, pode-se concluir que a utilização da metodologia proposta não resolve todos os problemas do universo escolar. Entretanto, ela é mais uma ferramenta que tem o intuito de contribuir para o êxito da aprendizagem dos alunos, ao mesmo tempo em que tende a diminuir a evasão escolar, pois os estudantes se sentem seduzidos pelos instrumentos tecnológicos nela presentes.

Espera-se, em trabalhos futuros, conseguir aplicar a metodologia “RobôBlocks” a professores para que possam avaliá-la em um viés educacional, pois não foi possível realizar esse tipo de estudo devido a pandemia da Covid-19. Espera-se, também, criar um livro de introdução à programação focado no desenvolvimento do pensamento computacional, que explore as possibilidades do *Kit* Educacional “RobôBlocks”.

REFERÊNCIAS

- ANDRADE, D.; CARVALHO, T.; SILVEIRA, J.; CAVALHEIRO, S.; FOSS, L.; FLEISCHMANN, A. M.; AGUIAR, M.; REISER, R. Proposta de atividades para o desenvolvimento do pensamento computacional no ensino fundamental. **Anais do Workshop de Informática na Escola**, v. 1, n. 1, p. 169, 2013.
- ARAÚJO, A. L. S. O. de; SCAICO, P. D.; PAIVA, L. F. de; RABÊLO, H. de M.; SANTOS, L. de L.; PESSOA, F. I. R.; TARGINO, J. M.; COSTA, L. dos S. Aplicação da taxonomia de bloom no ensino de programação com scratch. **Anais do Workshop de Informática na Escola**, v. 1, n. 1, p. 31, 2013.
- ARDUINO. **Arduino.cc**. 2019. Disponível em: <<https://www.arduino.cc>>. Acesso em: 14 maio, 2019.
- AURELIANO, V. C. O.; TEDESCO, P. Avaliando o uso do scratch como abordagem alternativa para o processo de ensino-aprendizagem de programação. **XX Workshop sobre Educação em Computação**, p. 10, 2012.
- BANZI, M. **How Arduino is open-sourcing imagination**. 2012. Disponível em: <<https://www.youtube.com/watch?v=UoBUXOOdLXY>>. Acesso em: 22 set. 2019.
- BARR, V.; STEPHENSON, C. Bringing computational thinking to k-12: what is involved and what is the role of the computer science education community? **Acm Inroads**, ACM New York, NY, USA, v. 2, n. 1, p. 48–54, 2011.
- BECKER, K. Back to pascal: retro but not backwards. **Journal of Computing Sciences in Colleges**, Consortium for Computing Sciences in Colleges, v. 18, n. 2, p. 17–27, 2002.
- BENITTI, F. B. V. Exploring the educational potential of robotics in schools: A systematic review. **Computers & Education**, Elsevier, v. 58, n. 3, p. 978–988, 2012.
- BENNEDSSEN, J.; CASPERSEN, M. E. Abstraction ability as an indicator of success for learning computing science? **Proceedings of the Fourth international Workshop on Computing Education Research**, ACM, p. 15–26, 2008.
- BLIKSTEIN, P. O pensamento computacional e a reinvenção do computador na educação. **Workshops do Congresso Brasileiro de Informática na Educação**, v. 4, n. 1, p. 1464, 2008.
- BULHÕES, D. B.; BARBOSA, F. R.; VIANA, R. P.; VILLELA, M.; SANTOS, C. Q. O uso da computação desplugada no processo de ensino-aprendizagem de alunos do curso técnico em informática. **Simpósio Brasileiro de Informática na Educação - SBIE**, v. 30, n. 1, p. 932, 2019.
- CAMBRUZZI, E.; SOUZA, R. M. de. O uso da robótica educacional para o ensino de algoritmos. **Anais do V Encontro Anual de Tecnologia da Informação**, p. 40–47, 2014.
- CAVALCANTE, A.; COSTA, L. D. S.; ARAUJO, A. L. Um estudo de caso sobre competências do pensamento computacional desenvolvidas na programação em blocos no code. org. **Anais dos Workshops do Congresso Brasileiro de Informática na Educação**, v. 5, n. 1, p. 1117–1126, 2016.
- CÉSAR, D. **Projeto robótica livre**. 2010. Disponível em: <<http://softwarelivre.org/robotica-livre/projeto-robotica-livre>>. Acesso em: 27 jun. 2019.

CÉSAR, D.; BONILLA, M. Robótica pedagógica com soluções tecnológicas livres no cet cefet em itabirito-minas gerais-brasil. **Anais do XXVII Congresso da SBC**, XIII Workshop sobre Informática na escola, v. 1, n. 1, p. 240–247, 2007.

CODE. **Curso 2**. 2013. Disponível em: <<https://studio.code.org/s/course2>>. Acesso em: 18 set. 2019.

COSTA, M. V. L.; JACON, L. d. S. C. Apoio ao processo de ensino-aprendizagem de programação de computadores: relato de experiência utilizando a programação de robôs autônomos seguidores de linha. **ReDiPE: Revista Diálogos e Perspectivas em Educação**, v. 1, n. 1, p. 124–138, 2019.

CSTA. Computational thinking teacher resources. **National Science Foundation under Grant No. CNS-1030054**, 2011.

DENNING, P. J. A debate on teaching computing science. **Communications of the ACM**, ACM, v. 32, n. 12, p. 1397–1414, 1989.

DIÁRIO OFICIAL DA UNIÃO. Resolução nº 510, de 07 de abril de 2016. **Conselho Nacional de Saúde** — seção 1, p. 44–46, 2016.

EVANGELISTA, S. R. M. Modelo para programação visual de matrizes (mvm): uma nova abordagem para visualização, manipulação e programação de algoritmos matriciais. **Embrapa Informática Agropecuária. Relatório Técnico (INFOTECA-E)**, Campinas: Embrapa Informática Agropecuária., 2001. ISSN 1517-0330.

EXTRA. **Lego Mindstorms Education - EV3**. 2020. Disponível em: <<https://cutt.ly/egQZcv1>>. Acesso em: 25 out. 2020.

FERREIRA, C.; GONZAGA, F.; SANTOS, R. Um estudo sobre a aprendizagem de lógica de programação utilizando programação por demonstração. **Anais do XVIII Workshop sobre Educação em Computação**, XXX CSBC, Belo Horizonte, MG, Brasil, p. 981–990, 2010.

FILHO, D. A. M.; GONÇALVES, P. C. Robótica educacional de baixo custo: Uma realidade para as escolas brasileiras. **Anais do XXVIII Congresso da SBC**, Workshop sobre Informática na Escola, v. 1, n. 1, p. 264–273, 2008.

FILIPEFLOP. **filipeflop**. 2019. Disponível em: <<https://www.filipeflop.com/>>. Acesso em: 02 maio. 2019.

FRANÇA, R.; FERREIRA, V.; ALMEIDA, L. d.; AMARAL, H. d. A disseminação do pensamento computacional na educação básica: lições aprendidas com experiências de licenciandos em computação. **Anais do XXII Workshop sobre Educação em Computação**, XXXIV Congresso da Sociedade Brasileira de Computação, p. 1473–1482, 2014.

GALLO, S. Transversalidade e educação: pensando uma educação não-disciplinar. **O sentido da escola**, DP&A Rio de Janeiro, v. 2, p. 17–41, 2000.

GOMES, F. I. M. **Construindo conhecimento: utilização de robots na aprendizagem de funções**. Dissertação (Mestrado) — Universidade da Madeira, Funchal, Portugal, 2013.

GOMES, T. C.; MELO, J. C. de. App inventor for android: Uma nova possibilidade para o ensino de lógica de programação. **Anais dos Workshops do Congresso Brasileiro de Informática na Educação**, II Congresso Brasileiro de Informática na Educação, v. 2, n. 1, p. 620–629, 2013.

GRAMIGNA, M. R. **Jogos de empresa**. São Paulo: Pearson Prentice Hall, 2007. v. 2. 160 p. ISBN 978-85-7605-121-3.

HEMMENDINGER, D. A plea for modesty. **Acm Inroads**, ACM, v. 1, n. 2, p. 4–7, 2010.

IFNMG. **Conheça**. 2019. Disponível em: <<http://www.ifnmg.edu.br/ifnmg/conheca>>. Acesso em: 24 nov. 2019.

IFNMG - CAMPUS MONTES CLAROS. **Projeto Pedagógico do Curso Técnico em Informática Integrado ao Ensino Médio**. 2020. Disponível em: <<https://www.ifnmg.edu.br/cursos-moc/tecnicos?id=1854>>. Acesso em: 12 Fev. 2020.

JÚNIOR, J.; RAPKIEWICZ, C. E.; DELGADO, C.; XEXEO, J. A. M. Ensino de algoritmos e programação: uma experiência no nível médio. **XIII Workshop de Educação em Computação (WEI'2005)**, São Leopoldo, RS, Brasil, 2005.

JÚNIOR, N. M. F.; VASQUES, C. K.; FRANCISCO, T. H. A. Robótica educacional e a produção científica na base de dados da capes. **Revista Electrónica de Investigación y Docencia (REID)**, Universidad de Jaén, Andaluzia, Espanha, n. 4, p. 35–53, 2010. ISSN 1989-2446.

LEITE, L. S.; POCHO, C. L.; AGUIAR, M. d. M.; SAMPAIO, M. N. Tecnologia educacional: mitos e possibilidades na sociedade tecnológica. **Tecnologia Educacional**, Rio de Janeiro, Brasil, v. 29, n. 148, p. 38–43, 2000.

LIVRE UFRJ. **DuinoBlocks4Kids**. 2019. Disponível em: <<http://ginape.nce.ufrj.br/LIVRE/paginas/db4k/db4k.html>>. Acesso em: 18 set. 2019.

MAISONNETTE, R. A utilização dos recursos informatizados a partir de uma relação inventiva com a máquina: a robótica educativa. **PROINFO - Programa Nacional de Informática na Educação**, Curitiba, Paraná, Brasil, p. 35, 2002.

MARTINS, F. N.; OLIVEIRA, H. C.; OLIVEIRA, G. F. Robótica como meio de promoção da interdisciplinaridade no ensino profissionalizante. **Anais do Workshop de Robótica Educacional**, 2012.

MELO, C.; AZOUBEL, M. A.; PADILHA, A. A metodologia da robótica no ensino fundamental: o que dizem os professores e alunos. **III Simpósio Nacional ABCiber**, São Paulo: ESPM Campus Prof. Francisco Gracioso, 2009.

MEO. **Currículo para o ensino de ciência da computação nas escolas**. 2008. Disponível em: <www.edu.gov.on.ca>. Acesso em: 24 nov. 2018.

MERKOURIS, A.; CHORIANOPOULOS, K.; KAMEAS, A. Teaching programming in secondary education through embodied computing platforms: Robotics and wearables. **ACM Transactions on Computing Education (TOCE)**, ACM New York, NY, USA, v. 17, n. 2, p. 1–22, 2017.

OLIVEIRA, R. **A robótica na aprendizagem da matemática: um estudo com alunos do 8º ano de escolaridade**. Dissertação (Mestrado) — Universidade da Madeira, Funchal, Portugal, 2007.

PAPARIDIS, O.; FRANCO, M. Plataforma arduino como apoio ao ensino de programação no curso de técnico em informática integrado. In: SBC. **Anais do XXIV Workshop sobre Educação em Computação**. [S.l.], 2016. p. 328–337.

PAPERT, S. **Teaching children thinking (LOGO memo)**. Cambridge, Massachusetts, United States: Massachusetts Institute of Technology, 1971.

PAPERT, S. **A máquina das crianças: repensando a escola na era da informática**. Porto Alegre: Artmed, 1994.

PEARS, A.; SEIDMAN, S.; MALMI, L.; MANNILA, L.; ADAMS, E.; BENNEDSEN, J.; DEVLIN, M.; PATERSON, J. A survey of literature on the teaching of introductory programming. **ACM sigcse bulletin**, New York, United States, v. 39, n. 4, p. 204–223, 2007.

QUEIROZ, R. L.; SAMPAIO, F. F.; SANTOS, M. P. d. Duinoblocks4kids: utilizando tecnologia livre e materiais de baixo custo para o exercício do pensamento computacional no ensino fundamental i por meio do aprendizado de programação aliado à robótica educacional. **Revista Brasileira de Informática na Educação – RBIE**, v. 27, n. 2, 2019. ISSN 1414-5685.

RAMOS, F.; TEIXEIRA, L. da S. Significação da aprendizagem através do pensamento computacional no ensino médio: uma experiência com scratch. **Anais do XXI Workshop de Informática na Escola**, IV Congresso Brasileiro de Informática na Educação, v. 21, n. 1, p. 217–226, 2015.

RIBEIRO, P. C.; MARTINS, C. B.; BERNARDINI, F. C. A robótica como ferramenta de apoio ao ensino de disciplinas de programação em cursos de computação e engenharia. **Anais do Workshop de Informática na Escola**, XXII Simpósio Brasileiro de Informática na Educação, v. 1, n. 1, p. 1108–1117, 2011.

RUSK, N.; RESNICK, M.; BERG, R.; PEZALLA-GRANLUND, M. New pathways into robotics: Strategies for broadening participation. **Journal of Science Education and Technology**, Springer, v. 17, n. 1, p. 59–69, 2008.

SANTOS, C. F.; MENEZES, C. S. de. A aprendizagem da física no ensino fundamental em um ambiente de robótica educacional. **Anais do XI Workshop de Informática na Escola**, XXV Congresso da Sociedade Brasileira de Computação, v. 1, n. 1, p. 2746–2753, 2005.

SCAICO, P. D.; HENRIQUE, M. S.; CUNHA, F. O. M.; ALENCAR, Y. M. de. Um relato de experiências de estagiários da licenciatura em computação com o ensino de computação para crianças. **Revista RENOTE**, v. 10, n. 3, 2012.

SILVA, A. F. d. **RoboEduc: Uma metodologia de aprendizado com Robótica Educacional**. Tese (Doutorado) — Universidade Federal do Rio Grande do Norte, Natal, Rio Grande do Norte, Brasil, 2009.

SILVA, E.; JAVARONI, S. L. Pensamento computacional e atividades com robótica para a promoção da aprendizagem sobre o significado do resto da divisão euclidiana. **Anais do XXIX Simpósio Brasileiro de Informática na Educação**, VII Congresso Brasileiro de Informática na Educação, v. 29, n. 1, p. 815–824, 2018.

SMITH, D. C. Building personal tools by programming. **Communications of the ACM**, Association for Computing Machinery, v. 43, n. 8, p. 92–92, 2000.

SOUZA, C. M. de. Visualg-ferramenta de apoio ao ensino de programação. **Revista Eletrônica TECCEN**, v. 2, n. 2, p. 01–09, 2009. ISSN 1984-0993.

TOCHÁČEK, D.; LAPEŠ, J.; FUGLÍK, V. Developing technological knowledge and programming skills of secondary schools students through the educational robotics projects. **Procedia-Social and Behavioral Sciences**, Elsevier, v. 217, p. 377–381, 2016.

VIDA DE SILÍCIO. **Portal Vida de Silício**. 2019. Disponível em: <<https://www.vidadesilicio.com.br>>. Acesso em: 16 set. 2019.

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

ZANETTI, H.; OLIVEIRA, C. Práticas de ensino de programação de computadores com robótica pedagógica e aplicação de pensamento computacional. **Anais dos Workshops do IV Congresso Brasileiro de Informática na Educação**, IV Congresso Brasileiro de Informática na Educação, v. 4, n. 1, p. 1236–1245, 2015.

ZILLI, S. R. **Apostila de robótica educacional**. Paraná: Expoente Informática - Gráfica Expoente, 2002.

APÊNDICE A – ATIVIDADE DESPLUGADA

Computação Desplugada

**Criado e adaptado para uso em sala de
aula por Rafael Porto Viana**

Atividade

Robôblocks—Resolução de Labirintos

Sumário

Os computadores armazenam dados em matrizes e vetores. O objetivo desta atividade é criar tabuleiros (matriz) e deslocar um objeto (robô) através de um labirinto, obedecendo uma sequência de instruções que estão vinculadas a sua localização dentro da matriz.

Matérias correlacionadas

- ✓ Programação: Matrizes e Vetores; Codificação e decodificação; Desenvolvimento de raciocínio e lógica.

Habilidades

- ✓ Contagem
- ✓ Raciocínio lógico
- ✓ Resolução de problemas
- ✓ Dar e seguir instruções

Idades

- ✓ A partir de 8 anos

Material

- ✓ Transparência feita a partir da folha “Transparência: Colorindo matriz com números” (página 4) e “Transparência: Percorrendo o Labirinto” (página 7)

Cada criança precisará de:

- ✓ Folha de Atividade: Crie seu próprio Labirinto (página 5)
Criando e Percorrendo o Labirinto (página 8)

Recorte as carinhas de emoticon e utilize para facilitar a resolução da atividade proposta na página 8.



Colorindo matriz com Números

Introdução

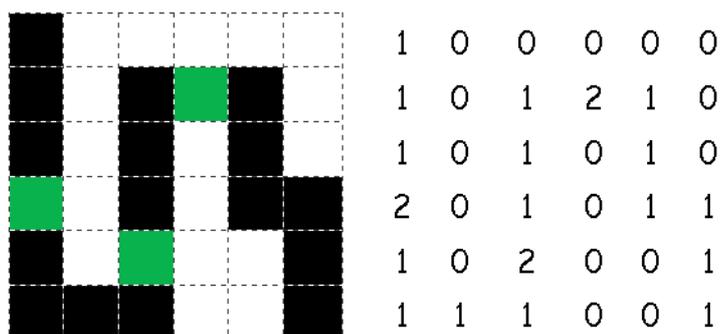
Questões para discussão

1. O que é uma matriz?
2. Como é feita a referência de uma posição dentro da matriz?

Criação de imagens com labirintos

As telas dos computadores são formadas por uma matriz de pequenos pontos chamados *pixels* (elementos de imagem). Em uma imagem em preto e branco, cada *pixel* é preto ou branco. Quando um computador armazena uma imagem, basta armazenar nos vetores da matriz quais pontos são pretos e quais pontos são brancos.

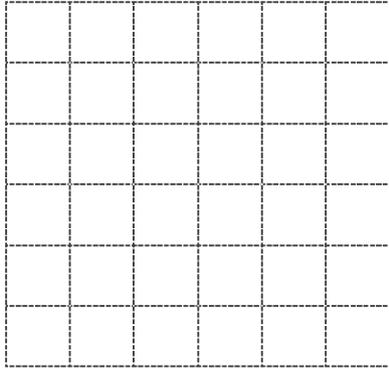
Suponha que o branco seja representado pelo “0”, o preto seja representado pelo “1” e o verde seja representado pelo “2”. A figura a seguir mostra como a imagem de um labirinto pode ser representada por números, referenciando o preenchimento de cada *pixel* de uma matriz.



A folha de atividade na página 4 apresenta algumas imagens que as crianças podem codificar e decodificar utilizando esse método.

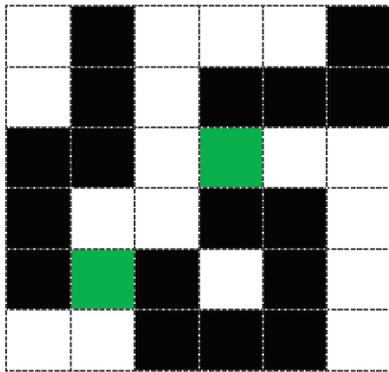
Transparência: Colorindo matriz com números

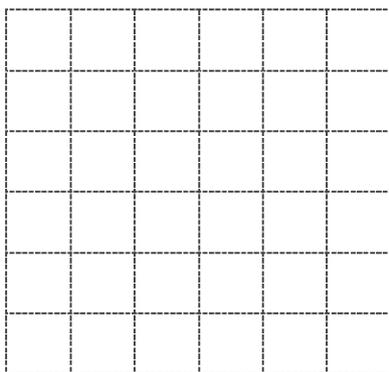
- Faça a decodificação dos números colorindo os *pixels* que compõem a imagem (matriz)



1	0	0	0	0	0
1	0	1	2	1	0
1	1	1	0	1	0
2	0	1	1	1	1
1	0	2	0	0	1
1	1	1	0	0	1

- Faça a codificação dos números colorindo os *pixels* que compõem a imagem (matriz)





- Grade em branco (para fins didáticos)

Folha de Atividade: Crie Seu Próprio Labirinto

Agora que você sabe como os números podem representar imagens em uma matriz, por que não tentar fazer seu próprio labirinto codificado para um amigo? Desenhe sua imagem na grade superior e, ao terminar, escreva os números de código ao lado da grade inferior. Recorte ao longo da linha pontilhada e dê o papel com a grade superior para um amigo codificar e a grade inferior para outro amigo decodificar. (Nota: o branco é representado pelo “0”, o preto pelo “1” e o verde pelo “2”).



Percorrendo o Labirinto

Introdução

Questões para discussão

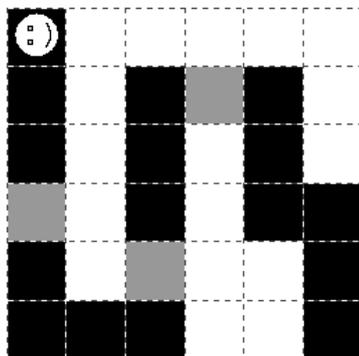
1. O que é uma instrução?
2. É adequado que pessoas sigam instruções à risca? Por exemplo, o que aconteceria se você apontasse para uma rua movimentada e dissesse “Atravesse a rua”?
3. Como as instruções são executadas pelos computadores?

Percorrendo a Matriz

Agora vamos percorrer a matriz seguindo instruções pré-determinadas. As instruções devem manter o robô dentro do labirinto, fazendo com que ele se desloque do início até o final do percurso. Quando o robô passar sobre o verde deve acionar uma buzina. As instruções existentes são:

				
Mover para frente	Mover para trás	Virar à Direita 90°	Virar à Esquerda 90°	Acionar Buzina

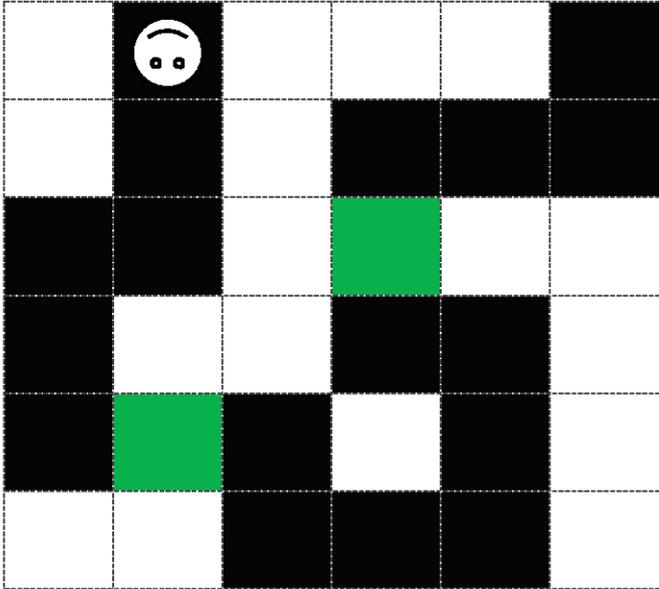
Exemplo: Nesse exemplo o robô representado pela carrinha de emoticon deve sair da posição (1,1) e chegar à posição (6,6) sem desviar do percurso proposto na matriz. Perceba que é necessário virar para direita antes de seguir para frente, pois o emoticon não está direcionado para onde deve seguir.



Posição	(1,1)	(2,1)	(3,1)	(4,1)	(5,1)	(6,1)	(6,2)	(6,3)	(5,3)	(4,3)
										
Instrução										
Posição	(3,3)	(2,3)	(2,4)	(2,5)	(3,5)	(4,5)	(4,6)	(5,6)	(6,6)	
										
Instrução										

Transparência: Percorrendo o Labirinto

- Indique a sequência de instruções para o robô percorrer o labirinto a seguir:



Posição										
Instrução										

Posição										
Instrução										

Posição										
Instrução										

Folha de Atividade: Criando e Percorrendo o Labirinto

- Pinte o tabuleiro de acordo com a numeração (0 = Branco, 1 = Preto, 2 = Verde), e indique a sequência de instruções para o robô percorrer o labirinto, iniciando na posição (1,5):

				(:)

1	1	1	0	1
<hr style="border: 0.5px solid black;"/>				
0	0	2	0	1
<hr style="border: 0.5px solid black;"/>				
1	1	1	0	2
<hr style="border: 0.5px solid black;"/>				
1	0	0	1	1
<hr style="border: 0.5px solid black;"/>				
1	2	1	1	0
<hr style="border: 0.5px solid black;"/>				

Posição	{									
Instrução	{									

Posição	{									
Instrução	{									

Posição	{									
Instrução	{									

APÊNDICE B – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO E QUESTIONÁRIO



MINISTÉRIO DA EDUCAÇÃO
Universidade Federal dos Vales do Jequitinhonha e Mucuri
Comitê de Ética em Pesquisa



TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE)

RobôBlocks

Você está sendo convidado(a) a participar de uma pesquisa intitulada: "RobôBlocks: uma metodologia de ensino para facilitar o aprendizado de programação", com o objetivo de investigar a sua opinião referente a esse projeto, que foi aplicado nas primeiras semanas de aula da disciplina de Introdução à Programação.

O projeto RobôBlocks propõe uma metodologia de ensino através de um kit de robótica educacional de baixo custo, para amenizar as dificuldades apresentadas por alunos iniciantes em programação. O Kit foi projetado para iniciantes em programação, mas pode ser aplicado a estudantes de qualquer área do ensino que não envolva a computação, visto que, destina-se ao desenvolvimento do pensamento computacional.

Sob orientação do professor Alexandre Ramos Fonseca (Email: arfonseca@ict.ufvjm.edu.br), a pesquisa é parte da dissertação de Rafael Porto Viana (Email: rafael.viana@ifnmg.edu.br), servidor do IFNMG e aluno do Programa de Mestrado Profissional em Educação, da Universidade Federal dos Vales do Jequitinhonha e Mucuri. Os dados coletados serão apresentados de forma agregada e utilizados na dissertação do curso de Mestrado em Educação do referido servidor. Os resultados desta pesquisa poderão ser apresentados em seminários, congressos e similares, entretanto, os dados/informações pessoais obtidos por meio da sua participação serão confidenciais e sigilosos, não possibilitando sua identificação.

A decisão em não participar da pesquisa não acarretará em nenhum tipo de constrangimento. Além disso, o participante poderá retirar seu consentimento a qualquer momento, sem qualquer tipo de prejuízo ou dano. A qualquer momento, o participante poderá entrar em contato com Rafael Porto Viana para fazer qualquer pergunta, que têm a obrigação de prestar os devidos esclarecimentos.

Desde já, agradecemos sua colaboração!

***Obrigatório**

1. Declaro que li os detalhes descritos neste formulário. Entendo que eu sou livre para aceitar ou recusar e que eu posso interromper minha participação a qualquer momento. Eu concordo que os dados coletados para o estudo sejam usados para os propósitos acima descritos. Por isso, concordo com o termo de Consentimento Livre e Esclarecido, bem como em participar desta pesquisa. *

Marcar apenas uma oval.

Concordo

Perfil Sociodemográfico

2. Sexo: *

Marcar apenas uma oval.

- Masculino
 Feminino
 Prefiro não responder

3. Você teve contato com alguma linguagem de programação antes de entrar no curso?

Marcar apenas uma oval.

- Sim
 Não
 Prefiro não responder

4. Quais linguagens de programação você costumava utilizar? *

Marque todas que se aplicam.

- C/C++
 C#
 Java
 JavaScript
 Python
 PHP
 Não programava antes do curso

Outra: _____

5. Você costumava utilizar a programação para quais finalidades antes de entrar no curso?

Marque todas que se aplicam.

- Trabalho
 Curiosidade
 Estudo
 Não programava antes do curso

Outra: _____

6. Você se sentiu motivado no decorrer das aulas usando o RobôBlocks? *

Marcar apenas uma oval.

- Sim
 Não
 Prefiro não responder

7. Você acha que seu desempenho escolar melhorou após a participação dessas aulas?

Marcar apenas uma oval.

- Melhorou
 Piorou
 Permaneceu igual
 Prefiro não responder

8. Você acha que sua frequência escolar melhorou após a participação dessas aulas?

Marcar apenas uma oval.

- Melhorou
 Piorou
 Permaneceu igual
 Prefiro não responder

9. Você acredita que obteve conhecimentos importantes para desenvolver um programa?

Marcar apenas uma oval.

- Sim, bastante
 Sim, um pouco
 Não, nenhum
 Prefiro não responder

10. Você recomendaria essas aulas para outro colega?

Marcar apenas uma oval.

- Sim
 Não
 Prefiro não responder

11. Você ficou satisfeito com essas aulas?

Marcar apenas uma oval.

- Sim
 Não
 Prefiro não responder

12. As seguintes habilidades e competências são estimuladas através da utilização do kit educacional RobôBlocks? Responda em uma escala de 1 a 4, sendo que 1 corresponde a "Nada", 2 a "Raramente", 3 a "Mais ou Menos" e 4 a "Frequentemente", *

Marcar apenas uma oval por linha.

	1	2	3	4
Raciocínio Lógico	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Trabalho em equipe	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Resolução de problemas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dar e seguir instruções	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Criatividade	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Leitura e compreensão	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Abstração e decomposição de problemas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Automação dos procedimentos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Aprendizado através dos erros	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Interdisciplinaridade	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Metodologia de Ensino

13. A metodologia de ensino proposta para aplicação do kit possui as seguintes características? *

Marcar apenas uma oval por linha.

	Sim	Não
Didática	<input type="radio"/>	<input type="radio"/>
Clareza	<input type="radio"/>	<input type="radio"/>
Sequencia Lógica	<input type="radio"/>	<input type="radio"/>
Objetividade	<input type="radio"/>	<input type="radio"/>

Agradeço a sua participação e colaboração.

14. Deixe seus comentários e/ou contribuições sobre a Metodologia de Ensino RobôBlocks:

ANEXO A – MANUAL DE INSTRUÇÕES

DuinoBlocks4Kids adaptado

Manual de Instruções para Instalação, Configuração e Uso

1. DB4K-SA

O DB4K-SA está disponível para Windows 32 bits e 64 Bits, e **roda localmente sem necessidade de acesso à Internet**. Porém, é necessário que o usuário tenha o software Arduino instalado na sua máquina.

O Kit educacional RobôBlocks utiliza a versão DB4K-SA adaptada. O download pode ser feito através do link <https://www.4shared.com/rar/BQbDt9Hqiq/DB4K-SA.html>.

2. Instruções de Instalação, Configuração e Uso

2.1 Instalando o Ambiente

DB4K-SA (Windows 32/64)

1. Baixe e instale o **Software Arduino 1.6.8**, <https://www.arduino.cc/en/Main/Software>
2. Baixe e descompacte o arquivo **DB4K-SA.rar** dentro da pasta C:
3. Copie o arquivo executável "**BD4K.exe**" de dentro da pasta **C:\DB4K-SA\arduexec** e cole na área de trabalho.

Feito isso, um ícone para execução do **DB4K-SA** ficará disponível na área de trabalho.

2.2 Configurando os Pinos de Entra e Saída

O DB4K vem com os pinos de entrada e saída utilizados pelos blocos **previamente configurados para o uso com o "RobôBlocks"**. No entanto, é **possível utilizar o DB4K para o controle de qualquer circuito que faça uso dos sensores e atuadores contemplados pelo DB4K**. Assim, caso se deseje modificar os pinos para o controle de circuitos montados com outra configuração de pinos e de entrada e saída, os números dos pinos podem ser reconfigurados. Toda essa configuração das pinagens, evidentemente, é feita pelo professor.

Caso deseje modificar os números dos pinos, edite os valores da seguinte maneira:

2.2.1 Modificando os pinos de entrada e saída no DB4K-SA

Os números dos pinos devem ser modificados no arquivo **DB4K-SA\ardublockly\db4k_setup.js** (Este arquivo pode ser editado utilizando-se programas como o Notepad++ ou o Bloco de Notas do Windows).

IMPORTANTE: Após modificar os valores dos pinos e salvar o arquivo de configuração, ao rodar o DB4K, é necessário fazer um "**reload**" da aplicação, de maneira que essas modificações sejam refletidas no código Arduino gerado pelo ambiente. Para tanto, entre no menu **<Configurações>** e selecione a opção **<atualizar pinos>**

2.2.2 Tabela com a configuração inicial dos Pinos do DB4K

A tabela abaixo apresenta os pinos previamente configurados no DB4K para o projeto RobôBlocks.

Componente	Pino	Componente	Pino
Motor DC Direito – sentido A	8	Ligar Motor DC Esquerdo	11
Motor DC Direito – sentido B	9	Sensor de Refletância Direita	A2
Ligar Motor DC Direito	10	Sensor de Refletância Esquerda	A3
Motor DC Esquerdo – sentido A	12	Sensor de Refletância Centro	A4
Motor DC Esquerdo – sentido B	13	Buzina	7

2.3 Criando um programa e enviando para a placa Arduino

2.3.1 Configurações Iniciais do Ambiente.

Estas configurações precisam ser feitas apenas na primeira vez que o ambiente for utilizado

Entre no menu <Configurações> e selecione a opção <Preferências>. Na Janela que abre, especifique:

1. A localização do Compilador Arduino.

DB4K-SA – Ao clicar em cima da linha para especificação do compilador, uma janela abrirá. Navegue até a pasta do Software Arduino (normalmente `C:\Program Files (x86)\Arduino` ou `C:\Program Files\Arduino`) e selecione o arquivo `Arduino_debug.exe`.

Exemplo:

- **Windows 64:** `C:\Program Files (x86)\Arduino\arduino_debug.exe`
- **Windows 32:** `C:\Program Files\Arduino\arduino_debug.exe`

2.3.2 Configurações da placa Arduino e da porta COM

Entre no menu <Configurações> e selecione a opção <Preferências>. Na Janela que abre, especifique:

1. A **Placa Arduino** desejada
2. A **Porta COM** a ser utilizada

Estas configurações têm que ser feitas sempre que você conectar uma placa Arduino no computador.

Obs. **Para se certificar de que o DB4K configurou a porta COM adequadamente**, clique sobre o nome da porta mesmo que ela já tenha sido definida automaticamente pelo DB4K.

2.3.3 Enviando um programa para a placa Arduino

Para carregar um programa na placa Arduino selecionada você deve montar o programa em blocos e apertar o botão laranja presente na parte superior da área de montagem de programas. Esse botão envia o programa diretamente para a placa Arduino configurada anteriormente.

Caso você deseje parar a execução do programa, aperte o botão vermelho presente na parte superior da área de montagem de programas. Este botão envia um “programa em branco” para a placa Arduino.

Enquanto o programa estiver sendo enviado, o botão de envio (ou parada), ficará cinza. É possível que, em algumas máquinas com Windows, o programa demore cerca de um minuto ou mais para ser carregado.

Existe uma área na parte inferior da janela do programa, chamada: Mensagens IDE Arduino, onde podem ser vistas as mensagens enviadas pela IDE.

2.3.4 Possíveis Problemas

1. Em algumas situações, é necessário que a **configuração da Placa Arduino e da porta COM seja feita antes no Software Arduino** para que as configurações dessas informações, dentro do DB4K, tenham efeito.
2. Caso, por alguma razão, o **DB4K não consiga enviar o programa para a placa**, pode-se utilizar a opção de menu <Código Arduino> <Abrir código na IDE Arduino> para abrir o código construído no Software Arduino e então enviá-lo para a Placa Arduino.

