

**UNIVERSIDADE FEDERAL DO PAMPA**

**RICARDO GOMES LOPES**

**INICIAÇÃO À PROGRAMAÇÃO EM BLOCOS UTILIZANDO ARDUBLOCK E  
PLATAFORMA ARDUINO**

**BAGÉ  
2019**

**RICARDO GOMES LOPES**

**INICIAÇÃO À PROGRAMAÇÃO EM BLOCOS UTILIZANDO ARDUBLOCK E  
PLATAFORMA ARDUINO**

Trabalho de Conclusão de Curso apresentado ao Curso de Licenciatura em Física da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Licenciado em Física.

Orientador: Pedro Fernando Teixeira Dorneles

Coorientador: Januário Dias Ribeiro

**BAGÉ  
2019**

Ficha catalográfica elaborada automaticamente com os dados fornecidos  
pelo(a) autor(a) através do Módulo de Biblioteca do  
Sistema GURI (Gestão Unificada de Recursos Institucionais) .

L864i Lopes, Ricardo Gomes  
Iniciação à programação em blocos utilizando Ardublock e  
plataforma Arduino / Ricardo Gomes Lopes.  
136 p.

Trabalho de Conclusão de Curso(Graduação)-- Universidade  
Federal do Pampa, FÍSICA, 2019.

"Orientação: Pedro Fernando Teixeira Dorneles".

1. Programação em blocos. 2. Ardublock. 3. Arduino. I.  
Título.

RICARDO GOMES LOPES

**INICIAÇÃO À PROGRAMAÇÃO EM BLOCOS UTILIZANDO ARDUBLOCK E  
PLATAFORMA ARDUINO**

Trabalho de Conclusão de Curso  
apresentado ao Curso de Licenciatura em  
Física da Universidade Federal do  
Pampa, como requisito parcial para  
obtenção do Título de Licenciado em  
Física.

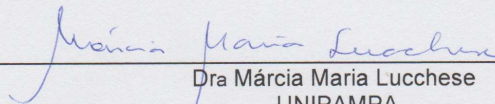
Trabalho de Conclusão de Curso defendido e aprovado em: 10 de julho de 2019.

Banca examinadora:



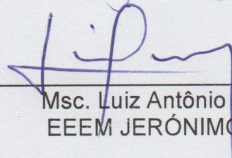
---

Msc. Januário Dias Ribeiro  
Coorientador  
UNIPAMPA



---

Dra Márcia Maria Lucchese  
UNIPAMPA



---

Msc. Luiz Antônio de Quadros Dworakowski  
EEEM JERÔNIMO MÉRCIO DA SILVEIRA

## **AGRADECIMENTO**

Agradeço aos meus familiares, em especial aos meus pais Francelina Camargo Gomes e Marcos Rinaldi de Melo Lopes, por nunca desistirem de mim e por terem me dado subsídios para ter uma vida pessoal e acadêmica de qualidade. As minhas irmãs Cristiane Gomes Lopes e Thais de Melo Lopes por me apoiarem. A minha namorada, Tércia Ferreira Machado, por ser uma pessoa incrível e ter me ajudado em momentos de dificuldade pessoal e por estar sempre ao meu lado.

Aos meus amigos e colegas, pelo companheirismo nos momentos bons e ruins.

Agradeço aos meus orientadores Dr. Pedro Fernando Teixeira Dorneles e Msc. Januário Dias Ribeiro pela orientação no presente trabalho, permitindo que o mesmo fosse concluído e por presarem sempre pela qualidade do mesmo. A todos os meus professores por terem me auxiliado no meu processo de ensino-aprendizagem, desde minha Educação Básica até o presente curso de Ensino Superior.

“Assim como toda felicidade é passageira,  
nenhum sofrimento será eterno.”

Desconhecido

## RESUMO

Este trabalho de Conclusão de Curso foi realizado com o objetivo de conceber a criação de um material instrucional com o enfoque na introdução à programação do *Arduino* utilizando o *software Ardublock*. A construção da primeira versão foi testada através de um estudo piloto, onde se utilizou como referencial de pesquisa Damiani (2013), de modo que, descreva a pesquisa através de uma intervenção pedagógica. A construção do material, seguiu como referencial a teoria da Aprendizagem Significativa de David Ausubel (2003). Os resultados obtidos nas intervenções pedagógicas permitiram a criação de uma nova versão do material instrucional, com correções e explicações extras de conceitos que não ficaram claras na intervenção, também inserindo novas temáticas no trabalho. Tal material tem como objetivo a introdução à programação do *Arduino* no Ensino Fundamental, nesta apostila contém a introdução à lógica de programação, comandos no *Ardublock*, divisão de tensão, sensores e atuadores e pequenos projetos envolvendo plataforma *Arduino/Ardublock*, de modo que, a apostila sirva como base para projetos mais avançados. Ainda, foram realizadas modificações no *Ardublock*, com a inserção de novas funções ao *software*, isso se deve em parte, por se tratar de um programa *opensource*, tais modificações foram realizadas com o objetivo de programar uma miniestação meteorológica, no qual o programa inicialmente não tinha funções suficientes para isto.

Palavras-Chave: Programação em blocos. *Ardublock*. *Arduino*.

## ABSTRACT

This work was completed with the purpose of designing the creation of an instructional material with the focus on the introduction to Arduino programming using *Ardublock* software. The construction of the first version was tested through a pilot study, where it was used as a research reference Damiani (2013), where it describes the research through a pedagogical intervention. The construction of the material, followed as a reference the theory of Significant Learning of David Ausubel (2003). The results obtained in the pedagogical interventions allowed the creation of a new version of the instructional material, with corrections and extra explanations of concepts that were not clear in the intervention, also inserting new themes in the work. This material has the objective of introducing Arduino programming in Elementary School, in this book contains the introduction to programming logic, commands in the *Ardublock*, voltage division, sensors and actuators and small projects involving Arduino / *Ardublock* platform, so that the work as a basis for more advanced projects. Also, modifications were made to the *Ardublock*, with the insertion of new functions to the software, this is due in part, because it is an opensource program, such modifications were made with the objective of scheduling a meteorological mini-session, in which the program initially did not had sufficient functions for this.

Keywords: Block programming. Ardublock. Arduino.



## LISTA DE FIGURAS

|  |    |
|--|----|
| Figura 1 – <i>Software Arduino IDE</i> e a extensão <i>Ardublock</i> . .....               | 28 |
| Figura 2 – Exemplo de passos de um algoritmo para acender uma lâmpada (Situação I). .....  | 32 |
| Figura 3 – Exemplo de passos de um algoritmo para acender uma lâmpada (Situação II). ..... | 33 |
| Figura 4 – Conjunto de todos Blocos Lógicos .....  | 34 |
| Figura 5.a – Exemplo 1: estrutura de condição “Se”. .....                                  | 35 |
| Figura 5.b – Exemplo 2 Estrutura de condição “Se” e “Senão”. .....                         | 35 |
| Figura 6 – Exemplo operador “e”. .....   | 36 |
| Figura 7 – Exemplo operado “ou”. .....   | 37 |
| Figura 8 – Exemplo de cartela da atividade do bingo lógico. ....                           | 37 |
| Figura 9 – Sorteio bingo lógico. ....  | 38 |
| Figura 10 – Aluna preenchendo cartela do bingo. ....                                       | 39 |
| Figura 11 - Demonstração de uma <i>protoboard</i> . ....                                   | 40 |
| Figura 12 – <i>Protoboard</i> /e suas conexões. ....                                       | 40 |
| Figura 13 – Explicação da <i>protoboard</i> utilizando a apostila. ....                    | 41 |
| Figura 14 – Condutores da <i>Protoboard</i> .....  | 42 |
| Figura 15 – Montagem do circuito da atividade do Blink .....                               | 43 |
| Figura 16 – Programação no <i>Ardublock</i> sendo realizada pelos alunos. ....             | 44 |
| Figura 17 – Abas principais do <i>Ardublock</i> . ....                                     | 45 |
| Figura 18 – Comandos Iniciais .....  | 45 |
| Figura 19 – Alteração do valor do <i>delay</i> . ....                                      | 46 |
| Figura 20 – Variáveis do <i>Ardublock</i> . ....   | 48 |
| Figura 21 – Diferenciação progressiva e reconciliação integrativa dos conteúdos. ..        | 54 |
| Figura 22 – Criação dos Blocos .....   | 56 |
| Figura 23 – Programação do Código em Java .....  | 57 |
| Figura 23 – Índice de resfriamento (sem modificação). ....                                 | 57 |
| Figura 24 – Índice de resfriamento (com modificação). ....                                 | 58 |

## LISTA DE QUADROS

|  |    |
|--|----|
| Quadro 1 – Lista de Revistas com seus endereços eletrônicos e amostragem temporal na qual se realizou a pesquisa. .... | 16 |
| Quadro 2 – Relação de artigos encontrados, identificados por autor(es), ano de publicação e título. ....               | 17 |
| Quadro 3 – Relação de artigos encontrados por revista. ....  | 22 |
| Quadro 4 – Duvidas, Facilidades e Dificuldades. ....   | 51 |
| Quadro 5 – Alterações na apostila. ....  | 52 |

## SUMÁRIO

|  |           |
|--|-----------|
| <b>1 INTRODUÇÃO .....</b>  | <b>12</b> |
| <b>2 OBJETIVOS.....</b>  | <b>15</b> |
| <b>2.1 Objetivo Geral.....</b>   | <b>15</b> |
| <b>2.2 Objetivos Específicos .....</b>                                       | <b>15</b> |
| <b>3 REVISÃO DE LITERATURA .....</b>   | <b>16</b> |
| <b>3.1 Revistas Pesquisadas.....</b>   | <b>16</b> |
| <b>3.2 Pesquisa.....</b>   | <b>17</b> |
| <b>4 REFERENCIAL TEÓRICO.....</b>  | <b>25</b> |
| <b>5 METODOLOGIA .....</b>   | <b>28</b> |
| <b>5.1 Arduino e <i>Ardublock</i>.....</b>                                   | <b>28</b> |
| <b>5.2 Estudo Piloto .....</b>   | <b>29</b> |
| <b>5.2.2 Local de aplicação .....</b>  | <b>29</b> |
| <b>5.2.3 Metodologia de Pesquisa .....</b>                                   | <b>29</b> |
| <b>6 RESULTADOS.....</b>   | <b>31</b> |
| <b>6.1 Estudo Piloto .....</b>   | <b>31</b> |
| <b>6.1.1 Estudo Piloto dia 12 de novembro de 2018.....</b>                   | <b>31</b> |
| <b>6.1.2 Estudo Piloto dia 26 de novembro de 2018.....</b>                   | <b>45</b> |
| <b>6.1.4 Achados em relação a intervenção sobre seus participantes .....</b> | <b>49</b> |
| <b>6.1.4 Achados em relação a intervenção propriamente dita: .....</b>       | <b>52</b> |
| <b>7 CONSIDERAÇÕES FINAIS .....</b>  | <b>59</b> |
| <b>REFERÊNCIAS.....</b>  | <b>61</b> |
| <b>APÊNDICE A – APOSTILA VERSÃO 1.1 .....</b>                                | <b>74</b> |

## 1 INTRODUÇÃO

A minha primeira experiência com a introdução da programação em blocos no Ensino Básico ocorreu no ano de 2015, quando fui bolsista do Programa Institucional de Bolsa de Iniciação à Docência (PIBID) Física da Unipampa – Campus Bagé, cujas atividades foram divididas em quatro etapas, que ocorreram durante aquele ano: i) introdução à lógica de programação e as estruturas de condição com uma atividade envolvendo um Bingo Lógico; ii) programação em blocos com o *software Scratch*; iii) apresentação da placa de prototipagem *Arduino* e programação; e iv) automação de experimentos a partir do *software S4A (Scratch For Arduino)*. Um segundo contato foi durante o componente curricular de Instrumentação para o Ensino de Física III, na qual se utilizou a programação do *Arduino IDE* (Software Oficial do Fabricante) e o uso do sensor Sonar.

Nessas experiências anteriores foi possível observar o potencial em despertar o interesse dos alunos pela interação com experimentos e interfaces de programação. No entanto, atividades dessa natureza ainda são pontuais e não estão presentes no cotidiano das escolas de Ensino Básico, apesar do crescente número de publicações nas principais revistas de Ensino de Física sobre esse tema, conforme será abordado na próxima seção.

A utilização de tecnologias também se justifica pelo que preconiza uma das diretrizes gerais da Base Nacional Comum Curricular (BNCC), sendo ela:

[...] Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva [...] (BRASIL, 2017).

Dessa forma, este trabalho tem como enfoque a automação de experimentos voltados para o Ensino Fundamental com a utilização da plataforma *Arduino* e o *software Ardublock*. A primeira é uma plataforma de prototipagem *open source* e a segunda uma extensão de programação em blocos que tem uma interação direta com o *software Arduino IDE*.

Utilizou-se como base a dissertação de mestrado de RIBEIRO (2017), na qual aborda uma introdução à programação e à automação de experimentos com *Arduino*, tendo como público alvo um grupo professores do Ensino Básico. Em seu

trabalho RIBEIRO (2017) descreve onze recomendações para aqueles que desejam trabalhar com a plataforma do *Arduino*:

- i. Levantar conhecimentos prévios utilizando pré-testes ou questionários investigativos;
- ii. Realizar atividades preliminares sobre lógica e eletrônica quando os conhecimentos prévios não forem muito consistentes;
- iii. Garantir, quando possível, que em cada grupo tenha indivíduos com conhecimentos prévios complementares entre lógica de programação e eletrônica;
- iv. Procurar que todos integrantes do grupo participem ativamente de todas as atividades, de modo a partilhar diferentes conhecimentos e experiências. Por exemplo, em duplas, os participantes podem se dividir entre as atividades de programação e montagens de circuitos, trocando papéis, realizando tarefas que são mais familiares e aprendendo sobre aquelas que não possuem experiências;
- v. Propor instrumentos avaliativos ao longo da realização das atividades, para verificar o andamento das aprendizagens dos participantes e ao mesmo tempo propiciar um feedback sobre as dificuldades a serem superadas;
- vi. Quando um grupo ou indivíduo realizar uma atividade com êxito, as tarefas podem ser modificadas em alguns pontos com a intenção de intensificar o pensamento lógico, fazendo pensar de formas diferentes sobre mesmos conceitos;
- vii. O ideal é que os códigos sejam construídos desde o início de cada atividade de programação, melhorando através da prática as dificuldades iniciais enfrentadas nas retomadas entre uma tarefa e outra;
- viii. Os circuitos montados na protoboard devem ser compreendidos no mesmo nível dos códigos de programação para os erros de projetos sejam corrigidos. Como o correto funcionamento depende dessas duas etapas, não é suficiente ter um entendimento maior em apenas uma delas. Identificar as deficiências em cada etapa é fator essencial;
- ix. As atividades práticas devem predominar na maior parte, para que os participantes não se desmotivem pelo excesso de informações novas. (RIBEIRO, 2017, p. 107).

Este trabalho vai adaptar e complementar tais recomendações estabelecidas por RIBEIRO, entretanto, seu foco será a introdução à programação em blocos voltadas para o Ensino Fundamental.

Foi concebida uma primeira versão de uma apostila para a introdução à lógica e a programação no *Ardublock*, que foi aplicada em uma escola da zona rural de Dom Pedrito/RS. Foi feita uma avaliação deste material através do modelo de pesquisa de intervenção pedagógica proposta por Damiani (2013), através de realizações de intervenções pedagógicas de 6 horas/aula. Este modelo propõe uma avaliação dos efeitos da intervenção sobre os participantes e uma avaliação da intervenção propriamente dita. Através dos achados da intervenção sobre os alunos,

foi construído uma nova versão da apostila, com inclusão de novas temáticas e correção em tópicos em que os alunos apresentaram dificuldades.

A construção da apostila segue como seu referencial teórico, a teoria da Aprendizagem Significativa de Ausubel (2003), na qual expressa a ideia de que para que ocorra a aprendizagem significativa é necessário que o aprendiz tenha pré-disposição para aprender e um material potencialmente significativo concebido pelo professor.

A última parte do trabalho são as modificações do *software Ardublock*, com inclusão de novas funções que não constavam na versão que estava sendo utilizada. O objetivo dessas modificações é auxiliar na programação de uma miniestação meteorológica.

## 2 OBJETIVOS

### 2.1 Objetivo Geral

Conceber uma apostila de introdução à programação na plataforma *Arduino* para os anos finais do Ensino Fundamental, utilizando o *Ardublock* e acrescentando novas funções ao programa para possibilitar a programação de uma miniestação meteorológica.

### 2.2 Objetivos Específicos

- i. Desenvolver uma revisão da literatura nas principais revistas de ensino de ciências e realizar a categorização nas seguintes temáticas: introdução à programação; programação em blocos; componentes eletrônicos e sensores; e *Arduino*.
- ii. Realizar um estudo piloto com alunos do Ensino Fundamental sobre a introdução à programação utilizando a ferramenta *Ardublock* e através de um diário de bordo, analisar as principais dificuldades dos alunos, com o intuito de aprimorar a produção educacional (apostila).
- iii. Conceber uma apostila com os princípios da teoria da aprendizagem de Ausubel, ou seja, construir um material capaz de criar melhores condições para a aprendizagem significativa.
- iv. Utilizar as linguagens Java, XML e C++ para modificar o *Ardublock*, propiciando melhores condições para estudantes dos anos finais do Ensino Fundamental.

### 3 REVISÃO DE LITERATURA

Nesta seção se descreverá como foi realizada a revisão da literatura. As revistas, artigos encontrados, o período inicial e final dos anos adotados para cada revista, número de artigos encontrados e uma descrição dos artigos relacionados à programação em blocos.

#### 3.1 Revistas Pesquisadas

A revisão da literatura parte de publicações de revistas da área de Ensino de Física, sendo analisadas nove revistas nacionais e duas revistas internacionais, totalizando onze revistas. Tais revistas estão listadas no Quadro 1.

A pesquisa dos artigos relacionados ao tema de pesquisa ocorreu de duas maneiras: a primeira foi com a procura de palavras-chaves, fazendo uso das ferramentas de buscas dos próprios portais das revistas já mencionadas na subseção anterior. A segunda maneira foi através de busca mais aprofundada em cada volume das revistas a partir do ano de 2008 até 2018. As palavras-chave utilizadas foram: *Programação, Scratch, Ardublock, S4A, Arduino, Sensores, Acelerômetro, giroscópio, sonar, LM35 e LDR.*

Quadro 1 – Lista de Revistas com seus endereços eletrônicos e amostragem temporal na qual se realizou a pesquisa.

(continua)

| Revista  | Endereço Eletrônico:  | Anos:                |
|--|---|----------------------|
| Revista brasileira de ensino de ciência e tecnologia | <a href="https://periodicos.utfpr.edu.br/rbect">https://periodicos.utfpr.edu.br/rbect</a>                         | 2008-2018            |
| Caderno brasileiro de ensino de física               | <a href="https://periodicos.ufsc.br/index.php/fisica/index">https://periodicos.ufsc.br/index.php/fisica/index</a> | 2008-2018            |
| Experiências em ensino de ciências                   | <a href="http://if.ufmt.br/eenci/">http://if.ufmt.br/eenci/</a>   | 2006-2018            |
| A Física na Escola                                   | <a href="http://www1.fisica.org.br/fne/">http://www1.fisica.org.br/fne/</a>                                       | 2008-2012/ 2016-2018 |
| Revista brasileira de física tecnológica aplicada    | <a href="https://periodicos.utfpr.edu.br/rbfta">https://periodicos.utfpr.edu.br/rbfta</a>                         | 2014-2018            |



Quadro 1 – Lista de Revistas com seus endereços eletrônicos e amostragem temporal na qual se realizou a pesquisa.

(conclusão)

|   |   |           |
|---|---|-----------|
| Revista brasileira de física tecnológica aplicada | <a href="https://periodicos.utfpr.edu.br/rbfta">https://periodicos.utfpr.edu.br/rbfta</a> | 2014-2018 |
| ACTIO: Docência em Ciências                       | <a href="https://periodicos.utfpr.edu.br/actio">https://periodicos.utfpr.edu.br/actio</a> | 2016-2018 |
| Revista brasileira de ensino de física            | <a href="http://www.sbfisica.org.br/rbef/">http://www.sbfisica.org.br/rbef/</a>           | 2010-2018 |
| Latin-American Journal of Physics Education       | <a href="http://www.lajpe.org/">http://www.lajpe.org/</a>                                 | 2008-2018 |
| Enseñanza de las ciencias                         | <a href="http://ensciencias.uab.es/">http://ensciencias.uab.es/</a>                       | 2010-2018 |
| Investigações em ensino de ciências               | <a href="https://www.if.ufrgs.br/cref/ojs">https://www.if.ufrgs.br/cref/ojs</a>           | 2008-2018 |
| Ciência & Educação                                | <a href="http://www.scielo.br/ciedu">www.scielo.br/ciedu</a>                              | 2008-2018 |

Fonte: Autor (2018).

### 3.2 Pesquisa

Obteve-se um resultado de 82 artigos. No Quadro 2, os artigos encontrados estão identificados de acordo com temáticas envolvidas no presente trabalho, sendo elas: 1 – Introdução a programação; 2 – Programação em blocos; 3 – Sensores e componentes eletrônicos; 4 – *Arduino*.

Quadro 2 – Relação de artigos encontrados, identificados por autor(es), ano de publicação e título.

(continua)

| Artigos:   | Temática(s) |   |   |   |
|--|-------------|---|---|---|
|  | 1           | 2 | 3 | 4 |
| Uma proposta didática para abordar o conceito de temperatura a partir de situações, à luz da teoria dos campos conceituais de Vergnaud (GRINGS; CABALLERO; MORREIRA, 2008).    |             |   | X |   |
| A tecnologia do motor elétrico para o ensino de Eletromagnetismo numa abordagem problematizadora (PIRES; FERRARI; QUEIROZ, 2013).  |             |   | X |   |
| Robótica educativa livre no ensino de Física: da construção do robô à elaboração da proposta didática de orientação metacognitiva (TRETIN <i>et al.</i> , 2015).               | X           | X | X | X |
| Medidas das eficiências térmicas em aquecedores solares: uma alternativa complementar para o ensino de conceitos de física para estudantes das engenharias (PENNEREIRO, 2016). |             |   | X | X |

Quadro 2 – Relação de artigos encontrados, identificados por autor(es), ano de publicação e título.

|   | (continuação) |  |   |   |
|---|---------------|--|---|---|
| Espectroscopia para o Ensino Médio utilizando a placa <i>Arduino</i> (KELLY; ROCHA; GERMANO, 2017).   |               |  | X | X |
| Programação no ensino de matemática utilizando Processing 2: um estudo das relações formalizadas por alunos do ensino fundamental com baixo rendimento em matemática (SOUZA; YONEZAWA, 2018). | X             |  |   |   |
| Constante evolução da Tecnologia da Informação exige novo perfil dos jovens profissionais (BINI; RESENDE; SILVEIRA, 2009).  | X             |  |   |   |
| Uma Possível Associação entre Ciência e Tecnologia no Ensino de Ciências: o exemplo do telefone (SILVA, 2015).  |               |  | X |   |
| Atividade experimental como recurso para interação de alunos com transtornos específicos de aprendizagem em Física Moderna e Contemporânea (MONTEIRO; FILHO; GRESCZYSCZYN, 2017).             |               |  | X |   |
| Simulação da visão das cores: decodificando a transdução quântica-elétrica (LORETO; SARTORI, 2008).   |               |  | X |   |
| Novas tecnologias no estudo de ondas sonoras (CALVALCANTE, 2013).   |               |  | X | X |
| Controle Remoto: princípio de funcionamento (parte 1 de 2) (CALVALCANTE; RODRIGUES; BUENO, 2013).   |               |  | X | X |
| Projeto de um sensor de pressão manométrica para ensino de física em tempo real (ROCHA; GUADAGNINI, 2014).  |               |  | X | X |
| Controle Remoto: observando códigos com o <i>Arduino</i> (parte 2 de 2) (CALVALCANTE; RODRIGUES; BUENO, 2014).  |               |  | X | X |
| Acelerômetro eletrônico e a placa <i>Arduino</i> para ensino de física em tempo real (ROCHA; MARRANGHELLO; LUCCHESI, 2014).   |               |  | X | X |
| Maquete didática de um sistema trifásico de corrente alternada com <i>Arduino</i> : ensinando sobre a rede elétrica (VISCOVINI <i>et al.</i> , 2015).   |               |  | X | X |
| Aquecimento e resfriamento da água, aproximados à forma real (OLIVEIRA; ALVES; BARBOSA, 2016).  |               |  | X | X |
| Experimento de condução térmica com e sem uso /de sensores e <i>Arduino</i> (ROSA <i>et al.</i> , 2016).  |               |  | X | X |
| Avaliação do módulo da aceleração da gravidade com <i>Arduino</i> (GUAITOLINI <i>et al.</i> , 2016).  |               |  | X | X |
| Aparato educacional para estudo da queda livre com análise do movimento (BALDO <i>et al.</i> , 2016).   |               |  | X | X |
| Projeto de um calorímetro de relaxação para ensino de Física (ROCHA; SARAIVA; LUCCHESI, 2017).  |               |  | X | X |

Quadro 2 – Relação de artigos encontrados, identificados por autor(es), ano de publicação e título.

|   | (continuação) |   |   |   |
|---|---------------|---|---|---|
| Um instrumento alternativo ao estudo de pilhas recarregáveis via <i>Arduino</i> (MATHIAS; SILVA; SILVA, 2017).  |               |   | X | X |
| Um laboratório portátil de baixo custo: medição de g utilizando um pêndulo e a placa Raspberry Pi (ALMEIDA; DIAS; JULIÃO, 2018).  |               |   | X |   |
| Uma experiência didática com aquisição automática de dados no laboratório de física do ensino médio (SILVA; VEIT; 2006).  |               |   | X |   |
| Trabalhos trimestrais: uma proposta de pequenos projetos de pesquisa no ensino da física (MÜTZENBERG; VEIT; SILVEIRA, 2007).  |               |   | X |   |
| Avaliação dos trabalhos trimestrais com base na opinião dos alunos (MÜTZENBERG; VEIT; SILVEIRA, 2007).  |               |   | X |   |
| Instrumentação para o ensino de física moderna e sua inserção em escolas de ensino médio - relato de experiência (VICENTINI <i>et al.</i> , 2011).  |               |   | X |   |
| Uma atividade experimental de física por meio de investigação multimodal representacional (GOYA; LABURÚ, 2014).   |               |   | X |   |
| As mudanças ocorridas nos programas de ensino da física, os laboratórios didáticos de física e a inclusão de novas tecnologias no desenvolvimento dos experimentos remotos (OLIVEIRA <i>et al.</i> , 2016). | X             |   | X | X |
| Abordagem de tópicos de mecânica quântica no ensino médio partindo da aproximação com o cotidiano (BIAZUS; ROSA, 2016).   |               |   | X |   |
| Robótica nas aulas de matemática do ensino médio: uma proposta educacional e de baixo custo (SILVEIRA; COELHO; SANTOS, 2017).   | X             |   | X | X |
| Automatização de laboratórios didáticos de física geral - construção de uma maquete experimental automatizada para a determinação da constante de planck. (OLIVEIRA, 2018)                                  |               |   | X | X |
| Construção de um termômetro para fins didáticos (MOREIRA; CELESTE, 2012)  |               |   | X |   |
| Mecânica do acelerômetro de smartphones e tablets (VIEIRA; AGUIAR, 2016)  |               |   | X |   |
| Estudo das cores com <i>Arduino Scratch</i> e tracker (CALCANTE; TEIXEIRA; BALATON, 2016)   | X             | X | X | X |
| Utilizando um led como fonte de energia (ALVEZ; SILVA, 2008)  |               |   | X |   |
| Investigação sobre o funcionamento de um termômetro digital de baixo custo (PEDROSO; PIMENTA; ARAÚJO, 2014)   |               |   | X |   |
| A Física no ensino médio integrado: uma sequência didática sobre eletricidade com aplicação do <i>Arduino</i> . (FERNANDES; HARTMANN; DORNELES, 2014)   |               |   | X | X |
| Low cost MFC control unit using microcontroller (ESPINDOLA <i>et al.</i> , 2017)  |               |   | X | X |

Quadro 2 – Relação de artigos encontrados, identificados por autor(es), ano de publicação e título.

|  | (continuação) |   |   |   |
|--|---------------|---|---|---|
| Utilização do software <i>Scratch</i> para a aprendizagem de lançamentos de projéteis e conceito de gravidade no ensino fundamental. (ANJO; FREITAS; ANDRADE, 2016)                            | X             | X |   |   |
| A placa <i>Arduino</i> : uma opção de baixo custo para experiências de física assistidas pelo PC. (SOUZA <i>et al.</i> , 2011)   |               |   | X | X |
| Física com <i>Arduino</i> para iniciantes (CAVALCANTE; TAVOLARO; MOLISANI, 2011)   |               |   | X | X |
| Observando as marés atmosféricas: uma aplicação da placa <i>Arduino</i> com sensores de pressão barométrica e temperatura. (CARVALHO; AMORIM, 2014)  |               |   | X | X |
| Estações meteorológicas de código aberto: Um projeto de pesquisa e desenvolvimento tecnológico. (SILVA <i>et al.</i> , 2015)   |               |   | X | X |
| Medição de temperatura: O saber comum ignorado nas aulas experimentais (VILAR <i>et al.</i> , 2015)  |               |   | X | X |
| Sensores digitais de temperatura com tecnologia one-wire: Um exemplo de aplicação didática na área de condução térmica (AMORIM; DIAS; SOARES, 2015)  |               |   | X | X |
| Instrumentation for mechanical vibrations analysis in the time domain and frequency domain using the <i>Arduino</i> platform. (VARANIS <i>et al.</i> , 2016)                                   |               |   | X | X |
| Medida de g com a placa <i>Arduino</i> em um experimento simples de queda livre. (CORDOVA; TORT, 2016)   |               |   | X | X |
| Uso da plataforma <i>Arduino</i> e do software PLX-DAQ para construção de gráficos de movimento em tempo real. (DWORAKOWSKI <i>et al.</i> , 2016)  |               |   | X | X |
| Ilustração de incertezas em medidas utilizando experimentos de queda livre (COLUCCI <i>et al.</i> , 2013)  |               |   | X | X |
| Estatística de contagem com a plataforma <i>Arduino</i> . (PEREIRA; SANTOS; AMORIM, 2016)  |               |   | X | X |
| Investigação do fenômeno ilha de calor urbana através da utilização da placa <i>Arduino</i> e de um sítio oficial de meteorologia. (SANTOS; AMORIM; DEREZYNSKI, 2017)                          |               |   | X | X |
| Visualização da forma de onda e conteúdo harmônico da corrente elétrica alternada em eletrodomésticos. (DIONISIO; SPALDING, 2017)  |               |   | X | X |
| Desenvolvimento de um kit experimental com <i>Arduino</i> para o ensino de Física Moderna no Ensino Médio. (SILVEIRA; GIRARDI, 2017)   |               |   | X | X |
| Desenvolvimento de um aparato experimental de baixo custo para o estudo de objetos em queda: análise do movimento de magnetos em tubos verticalmente orientados (SZMOSKI <i>et al.</i> , 2018) |               |   | X | X |
| Sistema fotogate de seis canais analógicos para laboratórios didáticos de física. (CARVALHO; APOLINÁRIO; SOARES, 2017)   |               |   | X | X |

Quadro 2 – Relação de artigos encontrados, identificados por autor(es), ano de publicação e título.

|   |  |  | (continuação) |   |
|---|--|--|---------------|---|
| On mechanical vibration analysis of a multi degree of freedom system based on <i>Arduino</i> and MEMS accelerometers. (VARANIS; SILVA; MERELES, 2017) |  |  | X             | X |
| Desenvolvimento de um obturador de feixe óptico utilizando um disco rígido de computador (TOMAZIO; ROMERO; MENDONÇA, 2018)                            |  |  | X             | X |
| Audiotermômetro: um termômetro para a inclusão de estudantes com deficiência visual. (CORDOVA <i>et al.</i> , 2018)                                   |  |  | X             | X |
| O plano inclinado de Galileu: uma medida manual e uma medida com aquisição automática de dados (SOARES; BORGES, 2010)                                 |  |  | X             |   |
| Sistema fotodetector econômico para utilização em laboratórios de ensino e pesquisa. (GUITIERRE, 2017)  |  |  | X             |   |
| Aprimorando e validando um fotogate de baixo custo. (MACÊDO; PEDROSO; COSTA, 2018)  |  |  | X             |   |
| Using a smartphone acceleration sensor to study uniform and uniformly accelerated circular motions. (PALACIO <i>et al.</i> 2014)                      |  |  | X             |   |
| Uma visão diferenciada sobre o ensino de forças impulsivas usando um smartphone (JESUS; SASAKI, 2016)   |  |  | X             |   |
| Sensor sonar de movimento para ensino de Física experimental (ROCHA; GUADAGNINI, 2010)  |  |  | X             |   |
| Use of Computer-Based Data Acquisition to Teach Physics Laboratories: Case study Simple Harmonic Motion (AMARANI; PARADIS, 2010)                      |  |  | X             |   |
| Construcción de un prototipo para experimentos de mecânica (MORALES, 2010)  |  |  | X             |   |
| Obtención de las curvas de saturación de alcohol etílico (GONZÁLES; MARCIAS; ZAPATA, 2010)  |  |  | X             |   |
| Transferência de calor em um concentrador solar (SAPIÉN; CHAVARRIA, 2010)   |  |  | X             |   |
| Teaching general concepts about sensors and transfer functions with a voltage divider (MARTINS; VIANNA, 2011)   |  |  | X             |   |
| Los diodos láser en la Enseñanza de la Física en Ingeniería (LEMUS; SERRA; COLLAZO, 2011)   |  |  | X             |   |
| Projeto de um sensor eletrônico baseado em extensometria para medição de força (GUADAGNINI; ROCHA; BARLETTE, 2011)                                    |  |  | X             |   |
| Low-cost sensing to teach energy for everyone (ROSENBERG; CUFF, 2011)   |  |  | X             | X |
| Acondicionamiento térmico de aire usando energía geotérmica-ondas de calor (LANNELLI; GIL, 2012)  |  |  | X             |   |

Quadro 2 – Relação de artigos encontrados, identificados por autor(es), ano de publicação e título.

|  | (conclusão) |   |    |    |
|--|-------------|---|----|----|
| Propriedades de um acelerômetro eletrônico e possibilidades de uso no ensino de mecânica. (ROCHA; MARRANGHELLO, 2013)  |             |   | X  | X  |
| Design, construction and testing of a vibrometer (USIBE; ADIAKPAN; OBU, 2013)  |             |   | X  |    |
| Mathematical model of a separately excited DC motor powered by a solar array using external starter resistances (ALVARADOS; GALLEGOS, 2014)                      |             |   | X  |    |
| El uso de calculadoras con sensores en el aprendizaje de circuitos eléctricos (ESPINOZA; ZAVALA, 2014)   |             |   | X  |    |
| Development and assembly of Radio Frequency Module using components with SMD technology for educational purposes (XIMENES <i>et al.</i> , 2017)                  |             |   | X  |    |
| Diseño, implementación e impacto de prototipos experimentales para mejorar la enseñanza de la ley de Biot-Savart en estudiantes de ingeniería (ANDA; DIAS, 2017) |             |   | X  |    |
| Estudio de la caída libre utilizando diferentes técnicas experimentales (GERMÁN <i>et al.</i> , 2017)  |             |   | X  |    |
| Evaluación de la Incertidumbre en la medición en el laboratorio introductorio de Física (BORROTO; BECQUER; CALZADILLA, 2017)                                     |             |   | X  |    |
| Total por temática:  | 7           | 3 | 79 | 42 |

Fonte: Autor (2018).

Como se pode observar, na última linha do Quadro 2 foram encontrados apenas 7 artigos relacionados a introdução a programação, 3 artigos relacionados a programação em bloco, 79 artigos na utilização de sensores e componentes eletrônicos e 42 artigos relacionados com aplicações utilizando o *Arduino*. Notoriamente, os artigos sobre a introdução de programação assim como, a programação em blocos, é uma minoria nos resultados encontrados.

No Quadro 3, são apresentados os números de artigos encontrados em cada revista.

Quadro 3 – Relação de artigos encontrados por revista.

| Revista  | Nº de Artigos |
|--|---------------|
| Revista brasileira de ensino de ciência e tecnologia | 9             |
| Caderno brasileiro de ensino de física               | 14            |
| Experiências em ensino de ciências                   | 9             |
| A Física na Escola                                   | 4             |

(continua)

Quadro 3 - Relação de artigos encontrados por revista.

|   | (conclusão) |
|---|-------------|
| Revista brasileira de física tecnológica aplicada | 3           |
| ACTIO: Docência em Ciências                       | 1           |
| Revista brasileira de ensino de física            | 24          |
| Latin-American Journal of Physics Education       | 18          |
| Enseñanza de las ciencias                         | 0           |
| Investigações em ensino de ciências               | 0           |
| Ciência & Educação                                | 0           |
| <b>Total:</b>                                     | 82          |

Fonte: Autor (2018)

Conforme o Quadro 3, pode-se observar que as duas revistas com mais publicações encontradas foram Revista Brasileira de Ensino de Física e Latin-American Journal of Physics Education, correspondendo aproximadamente a 50% dos trabalhos encontrados. Porém, os três trabalhos relacionados com programação em bloco não foram encontrados nestas revistas. Assim, identificamos que esta temática não tem sido explorada nas revistas que apresentam o maior número de trabalhos sobre programação e automação de experimentos.

Como o foco do trabalho é a programação em blocos, a seguir serão descritos brevemente os trabalhos desta temática.

No artigo de Tretin *et al.* (2015) é descrito uma atividade experimental, na qual se utiliza da plataforma *Arduino* e da extensão *Scratch for Arduino (S4A)*. Os autores justificam a escolha desta plataforma, principalmente devido a seu baixo custo, de modo que, escolas públicas poderiam obter *kit's* experimentais sem a demanda de uma grande verba na aquisição destes, no que diz respeito à escolha do S4A para programação, é usado como argumento que não se espera que os alunos da educação básica interajam com uma linguagem de programação, deste modo, a programação em blocos que esta extensão utiliza se torna uma alternativa mais viável para os alunos. O referencial teórico deste artigo sugere uma abordagem metacognitiva, e na sua metodologia é feita a construção de um “carrinho”, no qual utiliza na construção: um *chassi* de acrílico, motores, *jumpers*, ponte-h, led's e a placa de prototipagem *Arduino*. O público alvo desta publicação são alunos do Ensino Médio, deste modo, foram feitas duas propostas didáticas, sendo estas: o estudo sobre a trajetória e estudo da velocidade média.

O artigo de Cavalcante, Teixeira e Balaton (2016), faz uma abordagem experimental de espectroscopia de baixo custo. Os autores, em um primeiro momento, descrevem os *softwares* e a plataforma que vai ser utilizada: Plataforma *Arduino*, Extensão *Scratch for Arduino (S4A)* e o *software* Tracker. Foi feita uma breve explicação sobre o funcionamento do LED RGB. Na parte metodológica, monta-se o experimento e se programa na extensão S4A. Posteriormente, através de um vídeo análise, calibra-se o Tracker e se realiza a coleta de dados. Também é feita um estudo teórico, na qual se aborda a Lei de Young.

No artigo de Anjo, Freitas e Andrade (2016) foi proposta uma atividade na qual se utilizaria o *software* de iniciação à programação *Scratch* e do jogo “*Angry Birds Space*”. Os alunos, inicialmente, interagiram com o jogo *Angry Birds Space*, explorando noções de ângulo, velocidade, movimentos na horizontal e vertical. Posteriormente os alunos foram divididos em grupos, questionados sobre o que seria um movimento oblíquo e utilizaram a programação em *Scratch* para descrever o lançamento de projéteis.

Apesar de ter sido encontrado um número reduzido de trabalhos que busquem uma linguagem mais acessível sobre a iniciação à programação, os três encontrados foram fundamentais para subsidiar o presente projeto. Os trabalhos apresentam propostas didáticas implementadas com estudantes da Educação Básica, cujos resultados sugerem a inserção do ensino de programação em bloco para alunos do Ensino Fundamental e/ou Médio.

No entanto, os artigos utilizam-se em suas aplicações a programação em *Scratch* ou na extensão S4A, enquanto que neste trabalho utilizou-se o *software* *Ardublock*, pois este contém o recurso dos estudantes visualizarem o código na interface IDE do *Arduino* ao montarem o código no *Ardublock* e enviarem para a IDE. Tal diferencial pode criar melhores condições para o estudante passar a programar diretamente na interface do *Arduino* (conforme será descrito na próxima seção).



## 4 REFERENCIAL TEÓRICO

O referencial teórico utilizado no presente TCC foi a Teoria da aprendizagem significativa de Ausubel (AUSUBEL, 2003), no que se diz respeito ao processo de ensino/aprendizagem.

Para Ausubel (2003) a aprendizagem significativa ocorre quando os significados de novos conceitos são relacionados de forma não literal e não arbitrária com os conhecimentos prévios dos aprendizes. Desta forma, o autor defende que o material instrucional deve:

[...] (1) estar relacionado de forma não arbitrária (plausível, sensível e não aleatória) e não literal com qualquer estrutura cognitiva apropriada e relevante (i.e., que possui significado 'lógico') e (2) que a estrutura cognitiva particular do aprendiz contenha ideias ancoradas relevantes, com as quais se possa relacionar o novo material [...] (AUSUBEL, 2003, p. 1).

Caso os dois itens, apresentados na citação anterior, forem satisfeitos Ausubel (2003) define que o material instrucional pode ser denominado de material potencialmente significativo.

No entanto, a necessidade de um material potencialmente significativo, não é a única condição para a ocorrência da aprendizagem significativa, pois o aprendiz deve manifestar predisposição para relacionar de maneira não arbitrária e não literal os conceitos presentes no material com os seus conhecimentos prévios. Assim, Ausubel (2003) apresenta duas condições mínimas para a ocorrência da aprendizagem significativa, sendo elas: predisposição para aprender (motivação) e o material ser potencialmente significativo.

Além das condições necessárias, Ausubel, destaca o papel dos conhecimentos prévios para a ocorrência da aprendizagem significativa, uma vez que esta ancora-se sobre os conhecimentos aprendidos anteriormente pelo aprendiz. Ausubel (2003) denomina os conhecimentos prévios de subsunçores, os quais se organizam de forma hierárquica, ou seja, dos conceitos mais relevantes, que ficam presentes por período maior na estrutura cognitiva, aos conceitos não tão relevantes ao aprendiz.

Para Ausubel (2013) a forma de interação entre os novos conceitos e os conceitos prévios é o que caracteriza a diferenciação entre aprendizagem mecânica e significativa. A aprendizagem por memorização ou aprendizagem mecânica, pode

se relacionar com uma estrutura cognitiva, entretanto, apenas de uma forma arbitrária e literal e tem como consequência: não se relacionar com nenhum subsunçor, são interiorizadas apenas tarefas de aprendizagem relativamente simples, as informações ficam retidas na estrutura cognitiva por um curto período de tempo (a menos que seja bem compreendida) e são altamente vulneráveis a materiais semelhantes. Enquanto que a aprendizagem significativa, como já foi descrita anteriormente, se relaciona de forma não arbitrária e não literal com os conhecimentos prévios dos aprendizes, ou seja, a diferença entre a aprendizagem por memorização da aprendizagem significativa é como se relaciona os conceitos novos com a estrutura cognitiva do indivíduo (arbitraria e literal *ou* não arbitrariedade e não literal). Porém, ao contrário do que pode parecer, estes dois tipos de aprendizagem não são opostos, e sim complementares um do outro, uma vez que a aprendizagem mecânica pode ser útil quando se estuda uma nova área de conhecimento, em que ainda não existem subsunçores e, assim, pode servir na criação destes, para posteriormente ancorarem novos conceitos, isto é, possibilitar a interação não literal e não arbitrária dos novos conceitos.

Para apresentar novos conteúdos aos aprendizes Ausubel sugere o princípio da diferenciação progressiva, em que consiste em explicar aos alunos o contexto geral do tópico de estudo, para posteriormente serem diferenciados os conceitos. Para isso:

Ausubel se baseia em duas hipóteses: a) é mais fácil para seres humanos captar aspectos diferenciados de um todo mais inclusivo previamente aprendido, do que chegar ao todo de suas partes previamente diferenciadas; b) a organização do conteúdo de uma certa disciplina, na mente de um indivíduo, é uma estrutura hierárquica na qual as ideias mais inclusivas estão no topo da estrutura e, progressivamente, proposições, conceitos e fatos menos inclusivos e mais diferenciados. (MOREIRA, 1982, p. 21).

Juntamente com a diferenciação progressiva, é necessário o que Ausubel chama de reconciliação integrativa, cujo princípio é: “[..] *explorar, explicitamente, relações entre proposições e conceitos, chamar a atenção para diferenças e similaridades importantes e reconciliar inconsistências reais ou aparentes.*” (MOREIRA, 1982, p. 21). Assim, posteriormente a diferenciação dos conteúdos, é necessária que seja feita uma reconciliação com os estudos anteriormente

realizados, para os alunos identificarem que os conceitos mais específicos se relacionam com conceitos mais gerais.

## 5 METODOLOGIA

### 5.1 *Arduino e Ardublock*

De acordo com o site oficial do *Arduino* (2018):

O *Arduino* é uma plataforma de prototipagem eletrônica open-source que se baseia em hardware e software flexíveis e fáceis de usar. É destinado a artistas, designers, hobbistas e qualquer pessoa interessada em criar objetos ou ambientes interativos. O *Arduino* pode sentir o estado do ambiente que o cerca por meio da recepção de sinais de sensores e pode interagir com os seus arredores, controlando luzes, motores e outros atuadores. O microcontrolador na placa é programado com a linguagem de programação *Arduino*, baseada na linguagem Wiring, e o ambiente de desenvolvimento *Arduino*, baseado no ambiente Processing. Os projetos desenvolvidos com o *Arduino* podem ser autônomos ou podem comunicar-se com um computador para a realização da tarefa, com uso de software específico (ex: Flash, Processing, MaxMSP).<sup>1</sup>

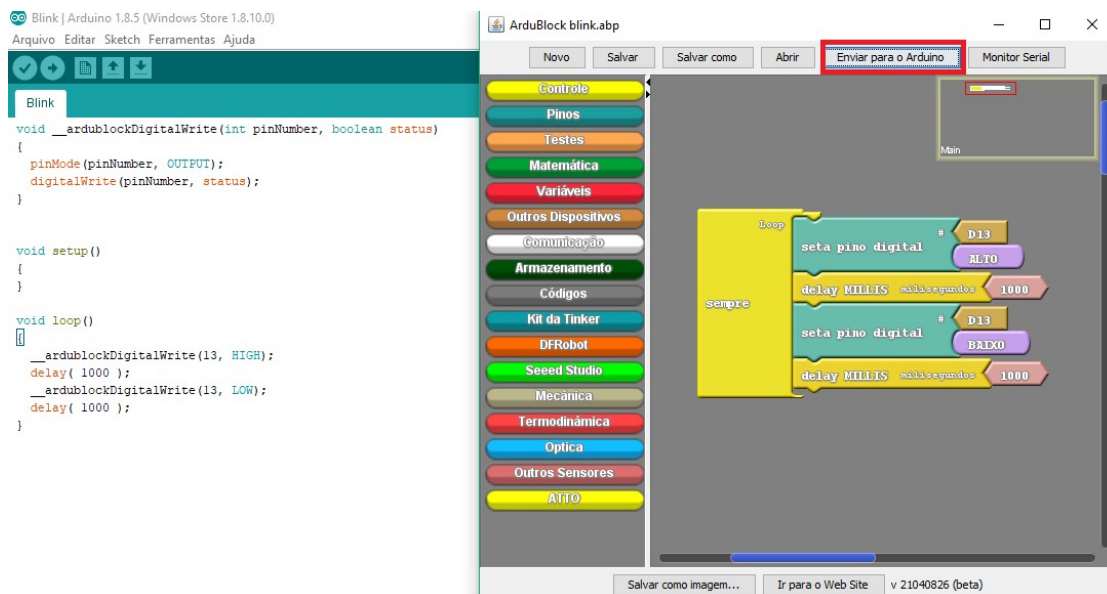
Além da citação acima, podemos descrever o *Arduino* como uma plataforma de testes para construção de projetos. A placa *Arduino* é projetada com um microcontrolador da Atmel AVR, com pinos digitais e analógicos para automação de projetos, seu conceito é de *hardware* e *software* livre, ou seja, permite que o usuário faça modificações e construa sua própria placa ou *software* baseado na plataforma do *Arduino*. Tais características tornam o seu custo mais barato e propiciam a criação de uma comunidade de usuários bastante ampla que se auxiliam em seus projetos tanto no que a construção dos circuitos quanto da programação.

O *Ardublock* é uma linguagem de programação construída em blocos com funções prontas, entretanto, é necessário carregar o programa na plataforma *Arduino*, ou seja, a linguagem em blocos vai ser transcrita em linguagem C++ (com pequenas modificações). O *Ardublock* (Figura 1) permite uma interação direta com a plataforma *Arduino*, deste modo, a programação realizada fica salva na placa mesmo após não estar mais conectada a um computador e a utilização do recurso *Plotter Serial* (construção de gráficos em tempo real) também fica disponível, assim como, o recurso *Monitor Serial* (Disponibiliza a visualização dos dados).

Figura 1 – *Software Arduino IDE* e a extensão *Ardublock*.

---

<sup>1</sup> Disponível em: <https://playground.arduino.cc/Portugues/HomePage>. Acesso em: 11 out. 2018.



Fonte: Autor (2018).

Na Figura 1 é possível observar a programação em blocos (lado direito) e a programação em linhas de comando (lado esquerdo) e a opção, destacada em vermelho, para enviar a programação feita no *ArduBlock* para o IDE do *Arduino*.

## 5.2 Estudo Piloto

Concluída a revisão da literatura, foi realizado um estudo piloto de 6 horas/aula dividido em dois encontros, utilizando uma primeira versão da apostila de introdução à programação em blocos no *Arduino/ArduBlock*.

### 5.2.2 Local de aplicação

As aplicações das atividades ocorreram no final do ano letivo de 2018, na Escola Municipal Rural de Ensino Fundamental Sucessão dos Moraes do Município de Dom Pedrito/RS, em uma turma do 8º ano do Ensino Fundamental, com 6 alunos, na disciplina de ciências.

### 5.2.3 Metodologia de Pesquisa

A metodologia de pesquisa adotada para avaliar a primeira versão da apostila foi a de uma intervenção pedagógica (DAMIANI, 2013), cujo seu caráter é de uma

pesquisa qualitativa. A utilização desta metodologia propiciou a avaliação dos objetivos específicos do Estudo Piloto, sendo eles:

- I. Analisar as principais facilidades e dificuldades que os alunos obtiveram em relação à lógica de programação e à plataforma *Arduino/Ardublock*;
- II. Registrar possíveis lacunas nas explicações didáticas da apostila;
- III. Avaliar o interesse dos alunos pelo tópico de estudo;
- IV. Adaptar e complementar as recomendações estabelecidas por Ribeiro (2017) para a introdução da programação em blocos, com foco no Ensino Fundamental.

Segundo Damiani (2013), a intervenção pedagógica é dividida em duas partes: o método da intervenção e o método da avaliação da intervenção. A metodologia da avaliação da intervenção analisa dois aspectos, sendo eles:

- a) *os efeitos que a intervenção teve sobre seus participantes,*
- b) *a avaliação da intervenção propriamente dita.*

Os instrumentos de coleta de dados utilizados para a avaliação sobre os efeitos sobre os participantes constituíram-se em:

- I. Diário de bordo para descrição e análise das observações realizadas;
- II. Gravações de áudios das respectivas intervenções;
- III. Respostas dos alunos nos guias de atividades.

Em relação à *avaliação da intervenção propriamente dita* foram analisados os objetivos específicos do Estudo Piloto (apresentados na subseção 2.2).

## 6 RESULTADOS

### 6.1 Estudo Piloto

Foi realizado um estudo piloto com intuito de testar uma primeira versão de um material instrucional. Foram feitos dois encontros de 3 horas/aula cada, totalizando um total de 6 horas/aula. Tais encontros têm como objetivo, o levantamento de dados, no qual o presente trabalho, usa como metodologia de pesquisa a intervenção pedagógica de Damiani (2013). As atividades foram realizadas na Escola Municipal de Ensino Fundamental Assumpção de Morais, localizada na zona rural do município de Dom Pedrito.

#### 6.1.1 Estudo Piloto dia 12 de novembro de 2018

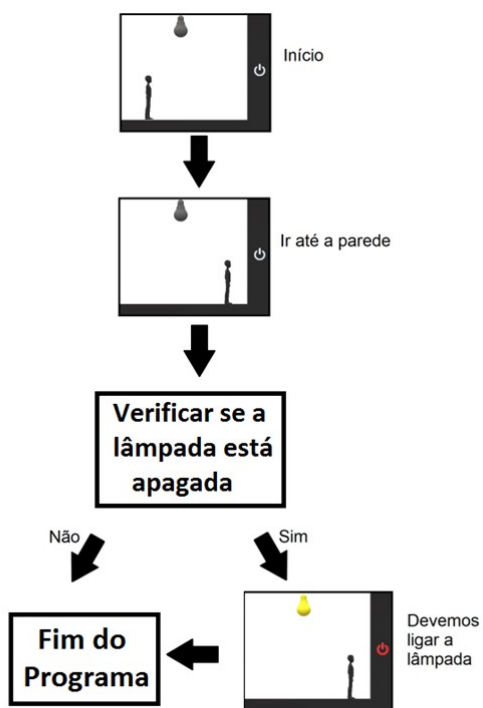
Inicialmente o autor deste trabalho se apresentou para turma. Estiveram presentes um total de 6 alunos, e perguntado a idade destes se constatou que as idades variavam entre 14 a 17 anos.

Na sequência, foi entregue a cada um dos alunos o primeiro capítulo do material instrucional (versão preliminar da apostila), sendo solicitado que eles guardassem seus aparelhos de celular e iniciou-se a intervenção pedagógica. Em um primeiro momento, o autor argumenta que um programa computacional/aplicativo nada mais é que a transcrição de uma linguagem humana para uma linguagem computacional e cita exemplos de dois programas que os alunos usam em seu cotidiano, sendo eles: *Messenger (Facebook)* e *WhatsApp*. Seguindo o conteúdo do material, foi explicado aos alunos que um programa segue uma sequência lógica e que isso é denominado de algoritmo. De forma interativa, são apresentados os passos que aparecem nas Figuras 2 e 3. Os passos foram: se aproximar do interruptor que liga as lâmpadas da sala de aula e verificar se as lâmpadas estão apagadas? Caso a resposta seja SIM, as lâmpadas deveriam ser acesas e o algoritmo encerrado em seguida. Caso contrário, não precisaria fazer nada e o algoritmo simplesmente deveria ser encerrado.

Em uma próxima atividade, foram repetidos os passos anteriores, com outra pergunta, se as lâmpadas estavam acesas? Caso a resposta fosse NÃO, eu deveria ligar a lâmpada e o algoritmo encerrado em seguida, senão simplesmente o algoritmo seria encerrado. Dessa forma, a parte inicial da apostila contém uma

abordagem simples do cotidiano, para os alunos compreenderem o que é uma sequência lógica e o que é um algoritmo.

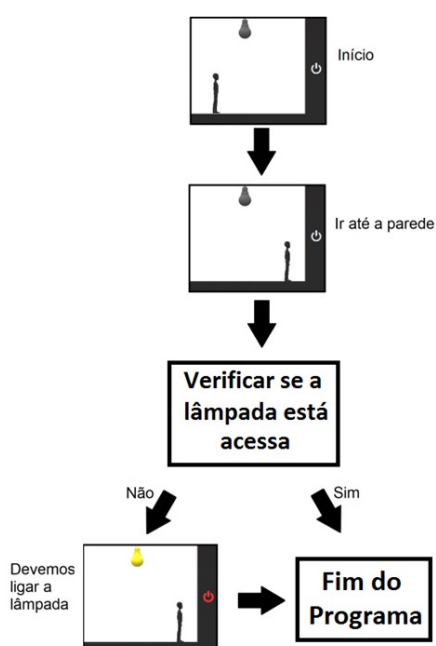
Figura 2 – Exemplo de passos de um algoritmo para acender uma lâmpada (Situação I).



Fonte: Autor (2018).



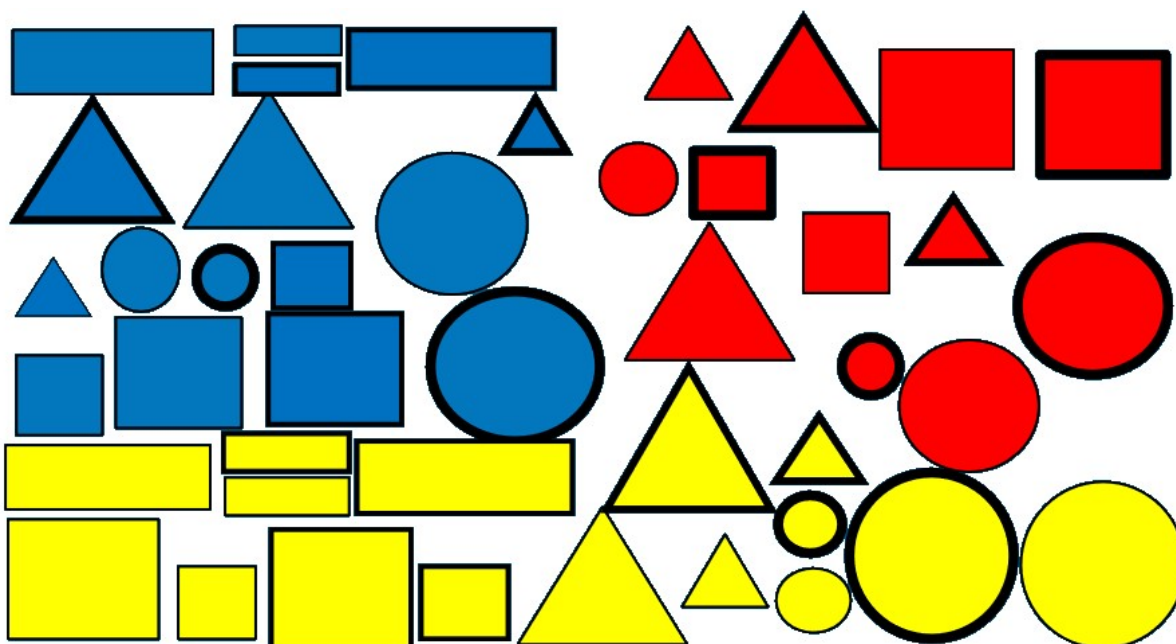
Figura 3 – Exemplo de passos de um algoritmo para acender uma lâmpada (Situação II).



Fonte: Autor (2018).

Posteriormente, utilizou-se o recurso de blocos lógicos (Figura 4) e foram apresentadas as características que um bloco lógico pode ter, sendo elas: forma (triângulo, retângulo, quadrado ou círculo), cor (azul, vermelho ou azul), tamanho (grande ou pequeno) e espessura da borda (grossa ou fina). Algo a se constatar que, ao contrário dos conceitos que tinham sido abordados até aquele instante, os alunos não compreenderam inicialmente o propósito dos blocos lógicos estarem sendo apresentado naquele momento, mas conforme será descrito nos parágrafos seguintes a atividade denominada “bingo lógico” contribuiu para uma melhor compreensão por parte dos alunos. Este tópico na apostila foi realocado para uma seção posterior à explicação dos operadores lógicos.

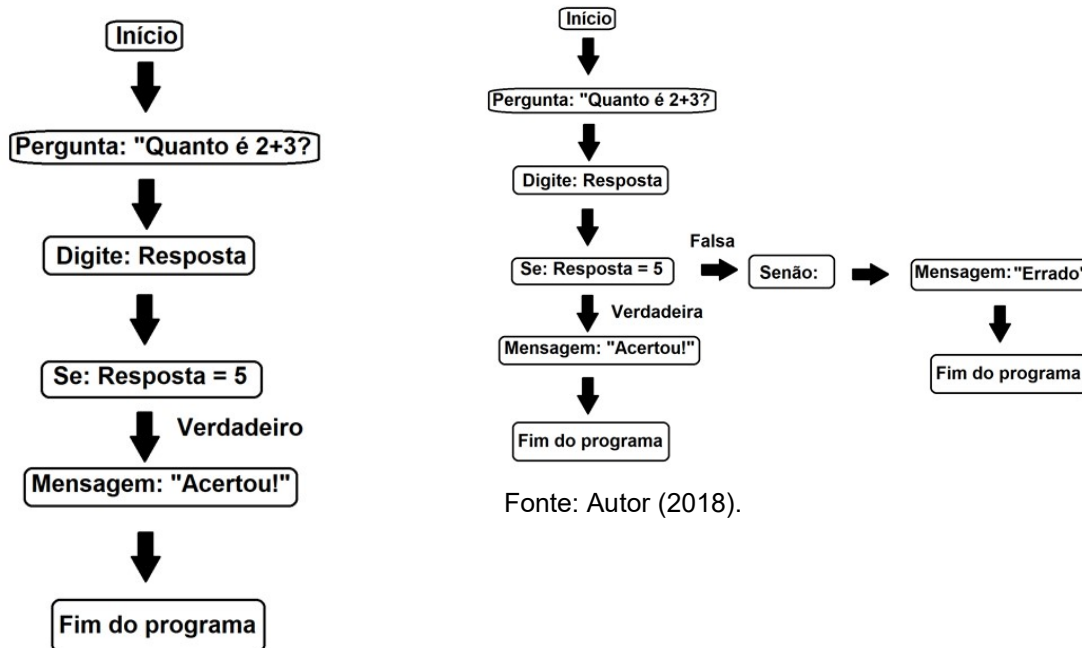
Figura 4 – Conjunto de todos Blocos Lógicos



Fonte: Programa Institucional de Bolsa de Iniciação à Docência – Universidade Federal do Pampa – Subprojeto Física (2017).

O próximo tópico de estudo foi as estruturas de seleção “se” e “senão”, explicando que, em tais estruturas, fazemos uma comparação de um parâmetro por outro. Para a estrutura de seleção “se” foi usado o seguinte exemplo: se fizermos uma pergunta, quanto é  $2+3$ ? Neste momento uma aluna respondeu que era igual a 5, o autor responde que a afirmação está correta e que a resposta que a aluna deu é um dos parâmetros e que o outro parâmetro é o número 5. A comparação entre os parâmetros será o “igual”, ou seja, se a resposta for igual a 5 (VERDADEIRO), teremos uma mensagem que a pessoa “Acertou!”, e que caso o usuário erre a resposta, simplesmente o programa vai encerrar. Para segunda estrutura de seleção “senão” foi usado como exemplo a mesma pergunta, os dois parâmetros e a comparador. Caso o usuário de uma resposta que seja diferente de 5 (FALSO), o programa não irá se encerrar e irá aparecer uma mensagem de que está “Errado!”. Podemos ver isso nas Figuras 5.a e 5.b.

Figura 5.a – Exemplo 1: Estrutura de condição “Se”.  
 Figura 5.b – Exemplo 2 Estrutura de condição “Se” e “Senão”.



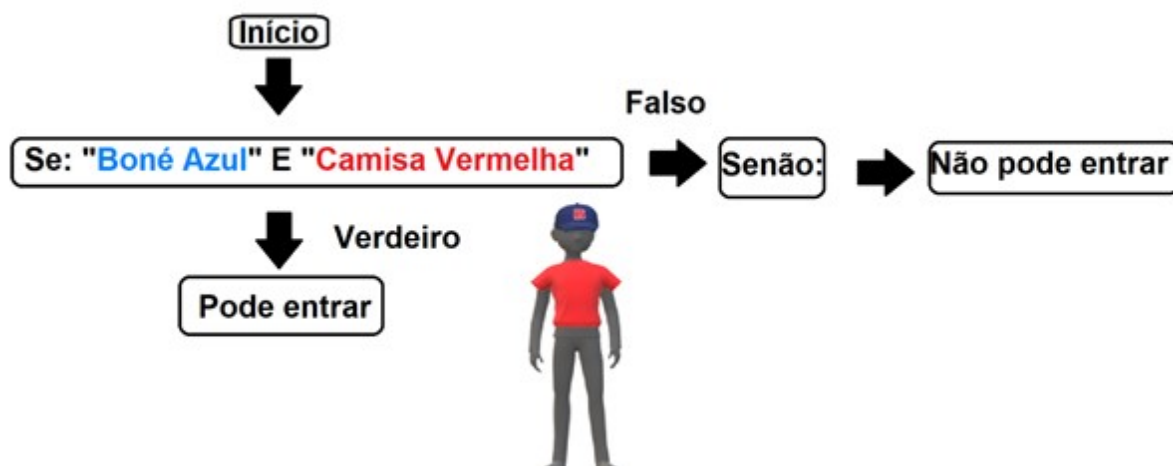
Fonte: Autor (2018).

Fonte: Autor (2018).

O autor, com o intuito de dar mais um exemplo, pergunta aos alunos se eles conheciam o “Show do Milhão” e eles responderam que sim. Então, foi dito a eles que existe um jogo para celular baseado no programa televisivo e que, seu funcionamento utiliza estruturas de seleção, em que temos uma pergunta e 4 opções de resposta. Porém, existe apenas uma resposta correta e que “se” eles escolhessem a resposta correta, o jogo iria emitir uma mensagem que eles acertaram, “*senão*” (caso escolhessem a resposta errada) o jogo iria emitir uma mensagem que eles erraram. Esse exemplo não consta na apostila, foi apenas mencionado de forma verbal.

O tópico seguinte foi os operadores lógicos “e” e “ou”, em que o operador “e” tem como função restritiva entre duas ou mais condições. Foi dado o exemplo de uma pessoa que deseja entrar em uma festa, mas para isso a pessoa precisa estar com um boné azul “e” uma camisa vermelha, assim sendo, a pessoa tem que estar com as duas peças de roupa para entrar, conforme pode ser visto na Figura 6.

Figura 6 – Exemplo operador “e”.

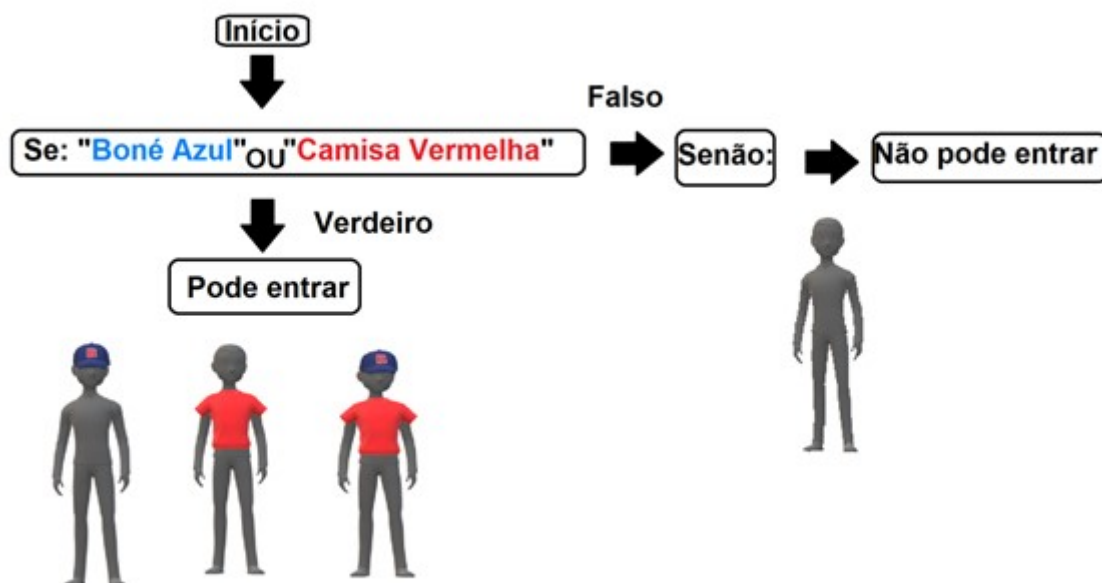


Fonte: Autor (2018).

O autor pergunta aos alunos se eles estavam compreendendo os conceitos abordados e uma aluna pergunta o aconteceria se as cores fossem ao contrário, ou seja, ou boné vermelho e a camisa azul. , Foi respondido a ela que a pessoa não entraria, pois para poder entrar o boné teria que ser azul e a camisa teria que ser vermelho, ou seja, as cores das roupas teriam que ser as mesmas da estrutura de condição, pois estas são as características que estão sendo analisadas, não só a peça de roupa mas também sua cor.

No entanto, se estivermos utilizando como operador lógico “ou”, por exemplo, uma camisa vermelha “ou” um boné azul, seria uma condição aditiva. Assim sendo, alguém que estiver com boné azul, ou com a camisa vermelha ou com os dois, poderá entrar na festa, pois se uma das condições for satisfeita, a sua entrada é permitida, podemos ver isso na Figura 7.

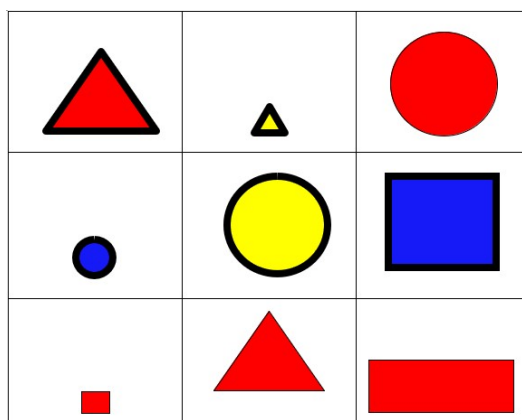
Figura 7 – Exemplo operado “ou”.



Fonte: Autor (2018).

Na sequência, foi trabalhada a primeira atividade da apostila, o “*Bingo Lógico*”. Foi distribuído a cada um dos alunos uma cartela do bingo com blocos lógicos, um exemplo de uma das cartelas pode ser visto na Figura 8:

Figura 8 – Exemplo de cartela da atividade do bingo lógico.

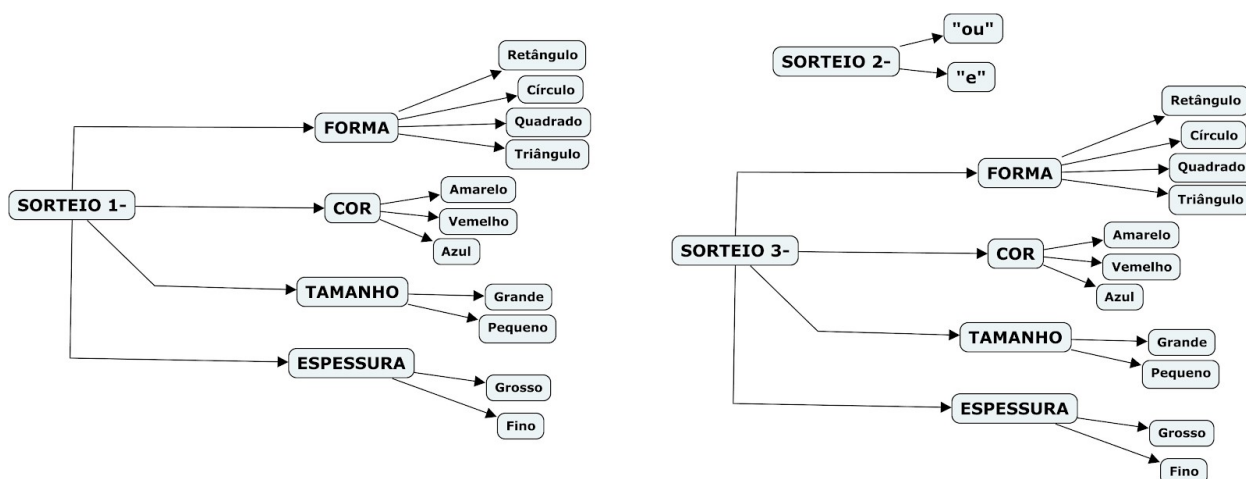


Fonte: Programa Institucional de Bolsa de Iniciação à Docência – Universidade Federal do Pampa – Subprojeto Física (2017).

Após a entrega das cartelas, foi explicado aos alunos as regras do bingo e como funcionaria seu sorteio, sendo feito tais sorteios com pedaços de papel.

Durante a atividade foi trabalhada apenas com uma estrutura de condição, sendo esta, a condição “Se”, pois para selecionar os blocos lógicos, precisavam de uma determina condição.

Figura 9 – Sorteio bingo lógico.



Fonte: Autor (2019).

A Figura 9 mostra três sorteios que devem ser feitos, sendo eles:

**SORTEIO 1** – O primeiro sorteio tem como finalidade de fazer o sorteio da primeira condição, ou seja, saber o primeiro parâmetro. O sorteio se divide em duas partes: 1 - Sortear um de quatro parâmetros: Forma, Cor, Tamanho ou Espessura; 2 - Sortear a especificação de um parâmetro, por exemplo: Se na primeira parte foi sorteado o parâmetro cor, devemos sortear posteriormente uma especificação se essa cor vai ser: Azul, Amarelo ou Vermelho.

**SORTEIO 2** – Este parâmetro serve para sortearmos o operador lógico “e”/”ou”.

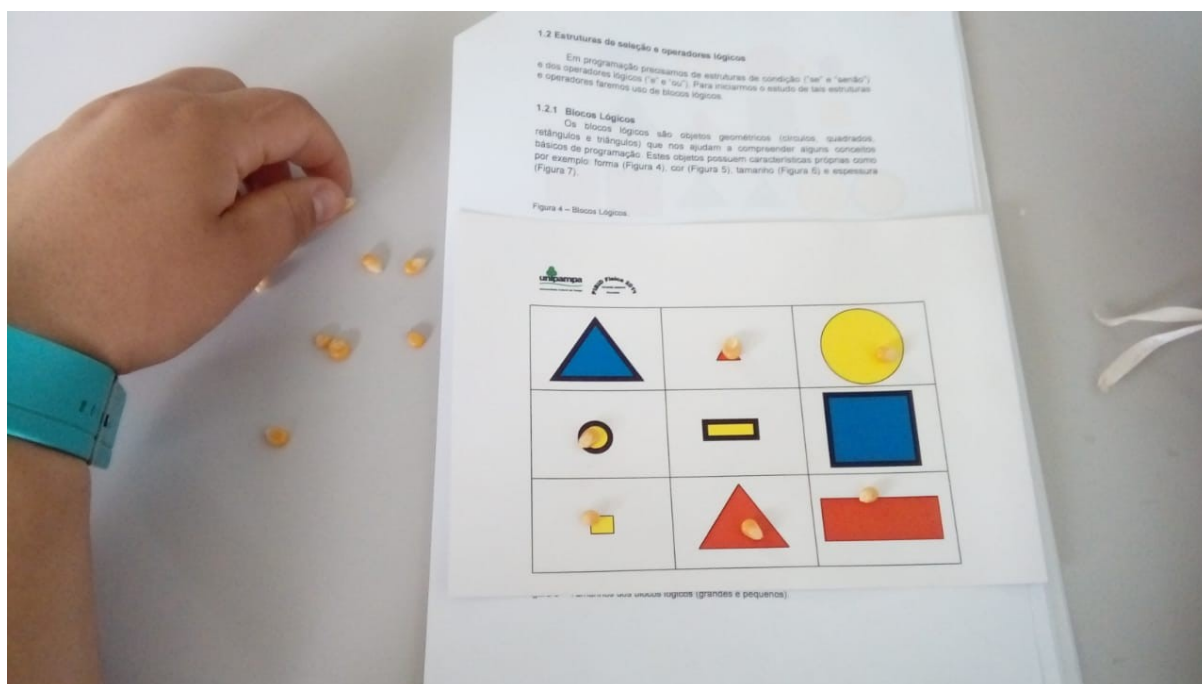
**SORTEIO 3** – Este sorteio serve para sortearmos o segundo parâmetro, o funcionamento do sorteio deste parâmetro é exatamente igual ao do sorteio 1.

Foi observado no momento da intervenção a necessidade demonstrar os conflitos que podem ocorrer. Observe o seguinte exemplo: caso seja sorteado como primeiro parâmetro a Forma: Círculo, no segundo sorteio foi sorteado o operador lógico “e” e o último sorteio a Forma: Quadrado. Existe um conflito claro, pois a estrutura corresponde será Se for círculo “e” e quadrado, perceba que não tem como

um bloco ser um círculo e um quadrado ao mesmo tempo. Deste modo, caso ocorra esse conflito, devemos realizar o sorteio 3 novamente. Tal necessidade foi adicionada na apostila, juntamente com exemplos de como funciona o bingo lógico.

Antes do bingo foram realizados dois sorteios para os alunos se habituarem à dinâmica do jogo. Ao longo da atividade os alunos interagiram de forma ativa e foi oferecido como recompensa uma caixa de bombom aos alunos que conseguissem preencher a cartela por primeiro. Na Figura 10, podemos ver uma aluna preenchendo a cartela do bingo lógico.

Figura 10 – Aluna preenchendo cartela do bingo.



Fonte: Autor (2018).

Na atividade foi perceptível que os alunos compreenderam bem o funcionamento da atividade do bingo lógico. Os conceitos que foram abordados até aquele instante, começaram a fazer sentido aos alunos.

Uma aluna pede para ir ao banheiro e o autor diz que vai esperá-la voltar para retomar a aula, enquanto se esperava, a fim de, instigar a curiosidade dos alunos foi descrito pelo autor as aplicações que a plataforma *Arduino* oferece, por exemplo, quando uma lâmpada irá acender ou apagar. Ainda se falou sobre os sensores de luminosidade, temperatura e o atuador LED (*Light Emitting Diode*). Foi mencionado



aos alunos que estes, iriam programar em blocos no *Arduino* e que um bloco encaixava no outro, e um aluno perguntou “- Tipo um quebra cabeça?” foi respondido que sim e que o mesmo realizou uma boa analogia.

Foi entregue uma placa *protoboard* para cada aluno e foi iniciada a parte da explicação sobre o seu funcionamento. O autor descreve que a *protoboard* é uma placa de testes e que é utilizada para montagem de circuitos elétricos, foi retirada a fita adesiva (que serve de isolante), para mostrar os condutores metálicos que ligam os terminais da placa, conforme podemos ver na Figura 11.

Figura 11 - Demonstração de uma *protoboard*.

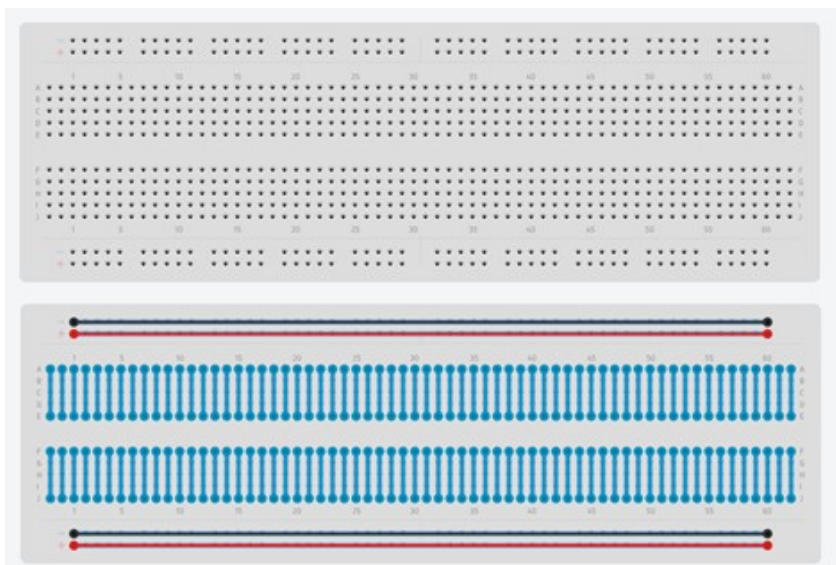


Fonte: Autor (2018).

Foi explicado o funcionamento das conexões de uma *protoboard* utilizando a Figura 12.

Figura 12 – *Protoboard* e suas conexões.

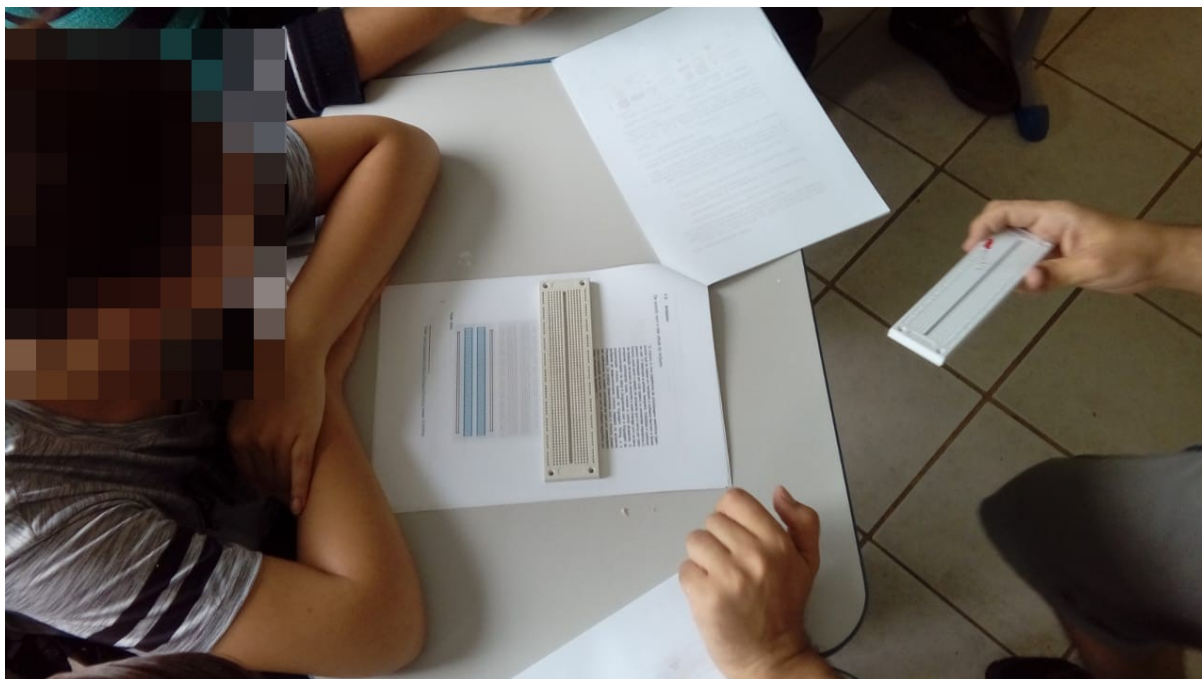




Fonte: Autor (2018).

Durante a explicação o autor pede para os alunos colocarem a *protoboard* em cima da imagem da apostila, enquanto era explicado seu funcionamento, conforme a Figura 13:

Figura 13 – Explicação da *protoboard* utilizando a apostila.

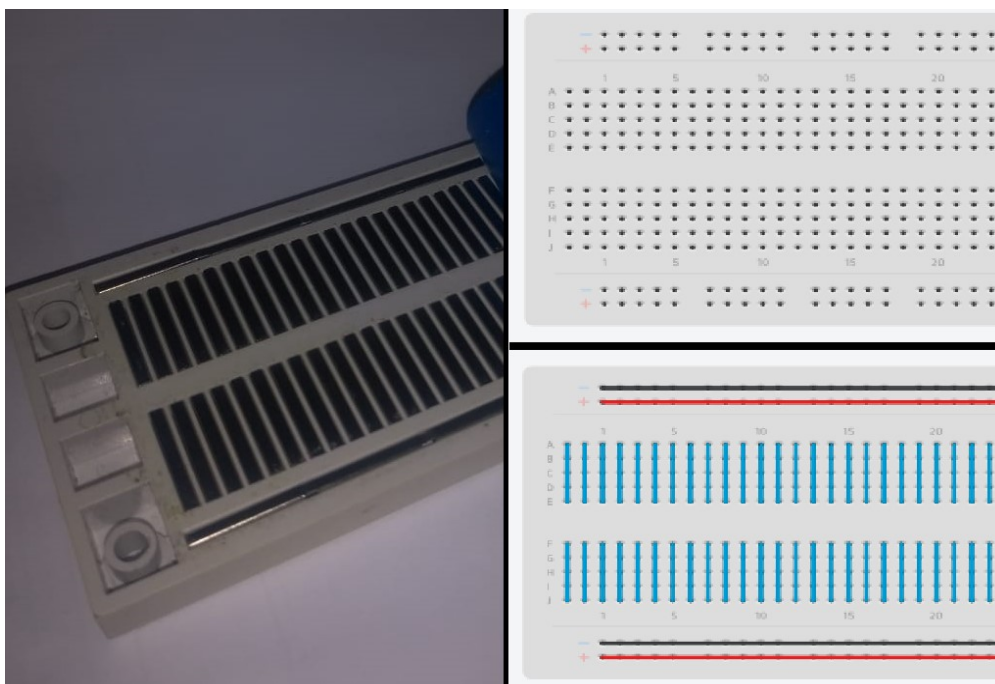


Fonte: Autor (2018).

Os alunos apresentaram uma preocupação, se poderiam levar choques com a *protoboard*, o autor explica que isso não iria acontecer.

É necessário constatar que, após aplicação desse estudo piloto, foi necessário realizar modificações em relação à explicação em relação ao funcionamento da *protoboard*, pois os alunos encontram dificuldade na segunda atividade da apostila “*Blink (Acender e apagar um LED)*” na hora de realizar as conexões dos terminais do *LED* e do resistor. Foi adicionada também uma fotografia dos condutores metálicos que ligam os terminais da *protoboard*, conforme pode ser visto na Figura 14:

Figura 14 – Condutores da *Protoboard*



Fonte: Autor (2019).

Os alunos foram divididos em grupos, pois havia apenas 3 computadores disponíveis. Uma aluna pergunta se ela poderia realizar programação a partir da aula que estava sendo ministrada, o autor respondeu que a programação na qual eles estavam aprendendo era com enfoque no *Arduino* e que ela teria que ter uma placa para poder programar. Devido à pergunta pertinente da aluna, foi informado aos alunos que devido à plataforma do *Arduino* ser *opensource*, ou seja, em código aberto existem vários fabricantes da placa e que é possível encontrar uma placa sendo vendida por US\$ 2 em sites chineses.

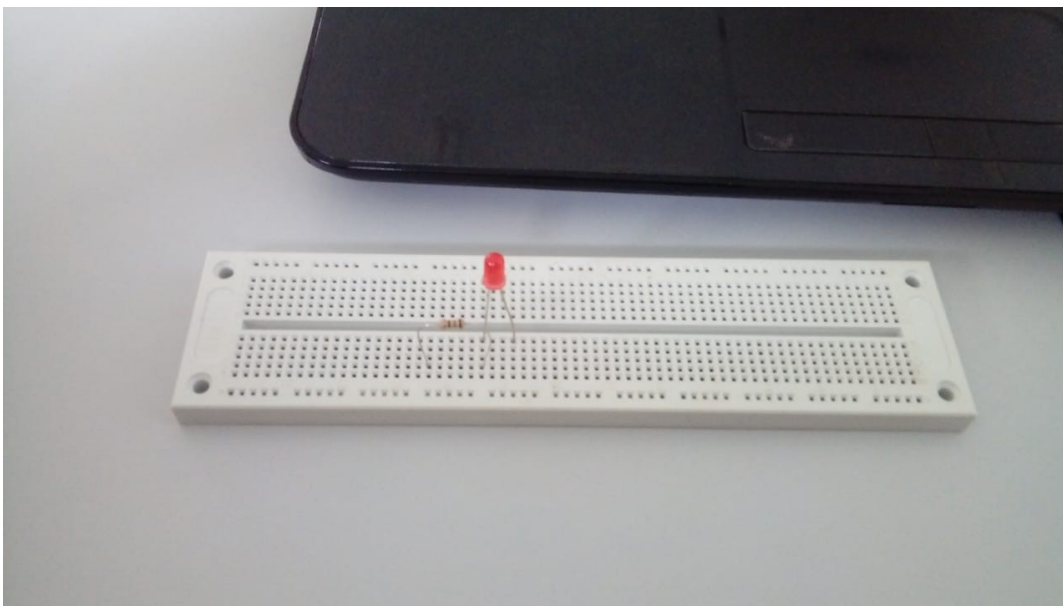
Após os alunos se organizarem em grupos, foi entregue a cada grupo uma placa *Arduino* UNO, e se explicou sobre o funcionamento dos seus pinos. Como este tópico ficou muito vago durante o estudo piloto, foi necessário acrescentar mais características da placa e de uma forma mais ilustrativa.

Explicou-se posteriormente o funcionamento do *Ardublock* e foi solicitado aos alunos que abrissem o programa, onde foram dadas as instruções sobre como abrir e como arrastar os blocos. Neste ponto é importante ressaltar, que não houve tantas dúvidas.

Apresentado o *Arduino* e o *Ardublock* se iniciou a segunda atividade intitulada de “*Blink (Acender e apagar um LED)*”. O autor explica aos alunos qual o programa eles devem utilizar para fazer um LED piscar. Para isso, o mesmo vai até o interruptor que acendem as lâmpadas da sala de aula e explica aos alunos que, se queremos fazer que a lâmpada pisque devemos acendê-la, esperar alguns segundos, apagá-la, esperar mais alguns segundos e acendê-la novamente. O autor explica que existem as portas digitais no *Arduino* e que temos dois estados possíveis: ALTO (ligado) e o BAIXO (desligado), ou seja, assim como o interruptor podemos colocar em um estado alto e um estado baixo. Posteriormente, foi explicado sobre o pino GND. Foi entregue aos alunos de cada grupo um LED e um resistor e dois *jumpers* e se discutiu o motivo do uso de uma resistência para se ligar uma lâmpada de LED.

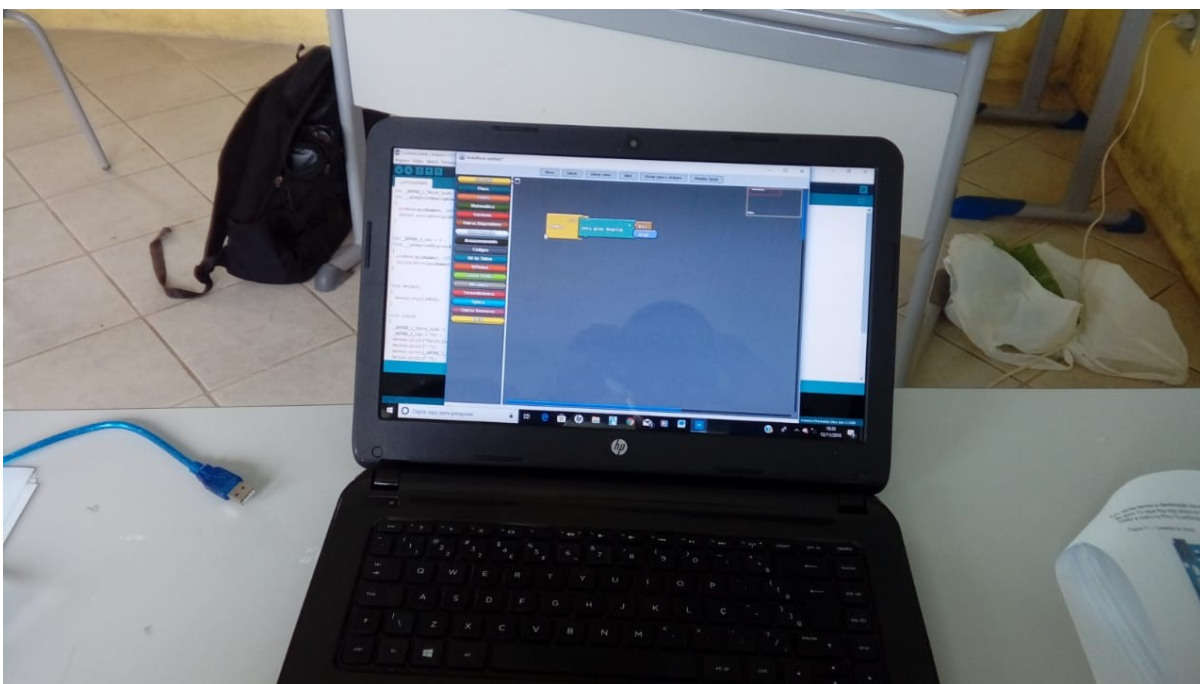
Os alunos deveriam nesta etapa conectar o LED e a resistor em série na *protoboard* e, posteriormente, ligar na placa *Arduino*. É necessário constatar que, os /alunos encontraram dificuldades em conectar os terminais dos dispositivos na *protoboard* e identificar posição correta para conectar o LED. Em um último passo, os alunos deveriam realizar a programação. Dois dos três grupos não encontraram dificuldades para realizar a programação. Entretanto, teve um grupo que apresentou mais dificuldades e o autor auxiliou estes na realização da programação. Todos os alunos conseguiram realizar a atividade. Nas Figuras 15, vemos o circuito montado pelos alunos e na Figura 16 a tela do computador de um grupo realizando a programação no *Ardublock*.

Figura 15 – Montagem do circuito da atividade do Blink



Fonte: Autor (2018).

Figura 16 – Programação no *Ardublock* sendo realizada pelos alunos.



Fonte: Autor (2018).

Assim, o primeiro encontro se conclui. Tal encontro foi importante para avaliar a motivação dos alunos e analisar as lacunas no material instrucional.

### 6.1.2 Estudo Piloto dia 26 de novembro de 2018

Foi entregue a cada um dos alunos uma parte do “*Capítulo III - Explorando Recursos*” da apostila, estiverem presente um total de 6 alunos. Nesta intervenção não houve necessidade da utilização da *protoboard*, entretanto, a placa *Arduino* foi utilizada.

Os alunos foram divididos em 3 grupos e foi entregue no início da intervenção um notebook para cada grupo e uma placa *Arduino*. Foi perguntado aos alunos se eles haviam trazido o material impresso entregue na aula passada (Capítulo II da apostila). Eles informam que sim, então o autor pediu para os alunos utilizarem o guia para abrir o *Ardublock* e acompanharem as abas de comando durante esta atividade.

Em um primeiro momento, iniciou-se mostrando as principais Abas do controle e foi informado aos alunos que iríamos estudar até a aba variáveis. Entretanto durante a aplicação de uma atividade, foi necessário a utilização da aba comunicação. Na Figura 17, podemos ver as principais abas do programa.

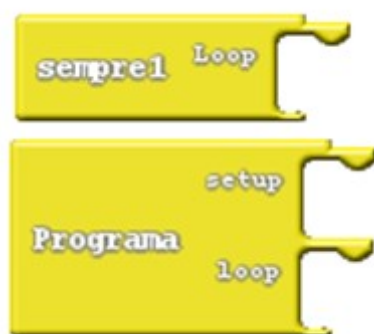
Figura 17 – Abas principais do *Ardublock*.



Fonte: Autor (2018).

A primeira aba ser trabalhada foi a aba “*Controle*” e foi demonstrada os dois primeiros comandos aos alunos, os comandos “*Sempre*” e comando “*Programa*” e foi explicado suas diferenças. Podemos ver estes comandos na Figura 18:

Figura 18 – Comandos Iniciais



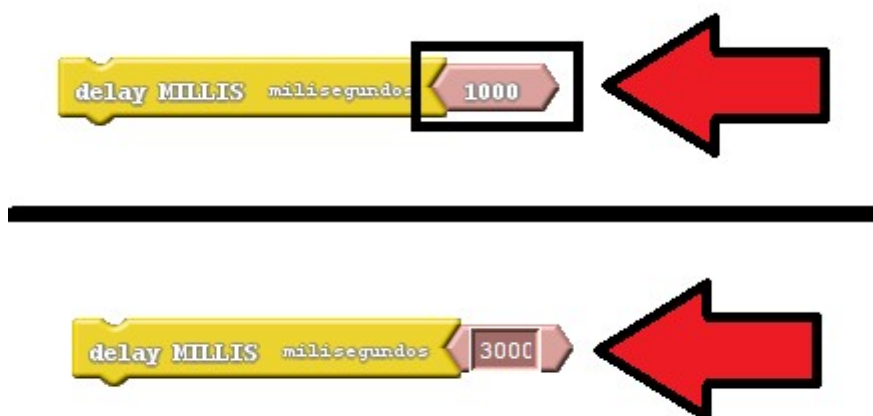
Fonte: Autor (2018).

O comando “*programa*” é utilizado quando desejamos definir parâmetros iniciais, foi dado o exemplo da aula anterior do código do *Blink*, em que era possível definir inicialmente se o LED estaria em um estado ALTO (Ligado) ou BAIXO (Desligado). Assim, se os parâmetros fossem colocados no “*Setup*” seriam executados uma única vez. Caso o programador não deseje definir parâmetros iniciais, ele poderia simplesmente usar o comando “*Sempre*”. Foi explicado aos alunos que, “*Loop*” significa laço e que os comando ali presentes iriam sempre se repetir após o programa ser executado.

Posteriormente foi apresentado os comandos de “*delay Millis*” e “*delay Micros*”, foi descrito que: este comando tem como função esperar um tempo para realizar o próximo passo. O autor pede para que os alunos peguem o comando de “*delay Millis*” e informa aos alunos que o bloco rosa conectado ao comando, pode ter seu valor alterado e pede para que os alunos mudem o valor que está dentro do comando.

Figura 19 – Alteração do valor do *delay*





Fonte: Autor (2019).

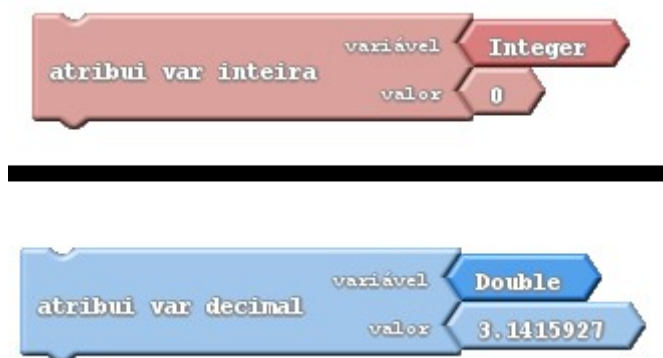
Foi explicado aos alunos que 1000 milissegundos é o que equivalente a 1 segundo, por meio de uma tabela de conversão o autor demonstrado aos alunos quantos milissegundos tem em 1 hora. Os alunos encontraram dificuldades nas conversões e a modificações se mostraram necessárias. Posteriormente, foi demonstrado aos alunos que 1000000 de microssegundos é o equivalente a 1 segundo um aluno questionou se, 1 microssegundo é menor que 1 milissegundo, foi informado a este, que seu raciocínio estava correto.

O comando “*tempo (ms)*” foi abordado, informando aos alunos que tal valor é uma variável que vai medindo o tempo transcorrido em milissegundos, isto é, esta variável vai calculando o tempo que está transcorrendo.

Posteriormente, foi explicado o que é uma variável e que no *Ardublock* temos dois tipos de variáveis: as inteiras e as decimais. Foi dado um exemplo matemático do que é uma variável, que na equação  $x-2=0$ , onde  $x$  é uma variável e seu valor é igual a 2. No *Ardublock*, atribuímos um valor a uma variável, dado o seguinte exemplo: atribuímos a uma variável inteira um valor igual a 3 e uma variável decimal um valor igual 3,2, um aluno disse que não conhecia o que era um valor decimal, foi explicado a ele que, um valor decimal é um número com virgula e perguntou a ele se ele já tinha visto tal conteúdo e ele informa que sim. A fim de esclarecer a dúvida do aluno foi dado o seguinte exemplo: R\$ 3 é um número inteiro e é igual a 3 reais, e que R\$ 3,2 é um número decimal, cujo seu valor é igual a 3 reais e 20 centavos. Foi explicado aos alunos que, devemos atribuir um nome a variável e sempre que desejarmos utilizá-la deveremos chamar pelo seu nome, para ficar mais claro esse

exemplo se inicia a atividade do cronômetro. Na Figura 20, vemos os dois tipos de variáveis do *Ardublock*.

Figura 20 – Variáveis do *Ardublock*



Fonte: Autor (2019).

Antes de começar a próxima atividade, o autor questiona os alunos sobre aparelhos que utilizamos para medir o tempo, os alunos respondem: relógio, temporizador, ampulheta e cronômetro. Foi informado de que, iremos construir um cronômetro utilizando o *Ardublock* e o *Arduino*

Os alunos encontraram dificuldades na hora de montar o programa, um dos grupos acabou escolhendo uma variável do tipo “inteira” ao invés da “decimal”, entretanto, tal erro serviu como aprendizado, uma vez que uma variável inteira não iria mostrar os décimos de segundos. Outro grupo encontrou dificuldade em chamar as variáveis com o mesmo nome que elas tinham dado anteriormente, mas o autor auxiliou este grupo e salientou sobre chamar a variável sempre pelo mesmo nome atribuído. Durante a atividade foram dadas dicas aos alunos, por exemplo: qual o valor que um cronômetro começa a medir? Os alunos responderam que em 0. Logo, estes deveriam estabelecer um valor inicial no *setup* igual a zero; o comando *tempo(ms)* mede o tempo em milissegundos, como fazemos para medir o tempo em segundos, os alunos responderam que devemos dividir o valor da variável por 1000, tal raciocínio estava correto. Durante a atividade notou-se que o ideal seria que esta atividade tivesse sido aplicada após o estudo da aba comando, pois esta se fez necessária, assim como, a aba matemática.

O próximo tópico de estudo foi as estruturas de seleção e os operadores lógicos, foi feita uma associação com a primeira intervenção e que, as estruturas



vistas anteriormente existiam no *Ardublock*. Nesta versão da apostila também continha uma tabela de comparadores. Os alunos não demonstraram dificuldades em relação a este tópico estudos.

Dando continuidade, foram explorados os comandos dos pinos digitais com estados ALTO e BAIXO e dos pinos analógicos. Em relação aos pinos digitais os alunos não encontraram dificuldades e fizeram uma associação com, entretanto, o autor reconhece que faltou um exemplo sobre os pinos analógicos.

O último tópico de estudos foi a aba matemática, apesar dos alunos terem visto este comando na atividade do cronômetro, foi feita uma abordagem mais formal. Foram demonstradas as 4 operações básicas no *Ardublock*: Soma, Subtração, Multiplicação e Divisão e que sua finalidade era realizar essas operações dentro de um programa.

Os alunos se mostraram muito motivados e pediram mais uma atividade, o autor então propõe aos alunos que criem um contador de números naturais inteiros em ordem crescente, e foram informados que isso tem uma aplicação quando queremos saber quando uma determinada ação aconteceu. Os alunos agora não apresentaram praticamente dificuldades para realização desta atividade, o autor com o intuito de instruir os alunos sobre a lógica de programação pergunta aos alunos: “ – *Se eu quiser fazer uma sequência que dê os número inteiro por exemplo, eu tenho o número 2 e quero chegar no número 3, tenho o número 3 e quero chegar no número 4, lembrando que dentro loop o comando sempre se repetirá?*” e um aluno responde: “- *Tem que ir somando/ de 1 em 1*”, foi confirmado que estava correto esse raciocínio. Esta atividade não consta na apostila e foi utilizada apenas porque os alunos demonstraram interesse em realizar mais uma atividade.

Durante as duas atividades, por não constar na apostila o comando de imprimir, o autor instruiu os alunos sobre sua utilização e que sempre que queremos que algo seja impresso no “*Monitor serial*” deveríamos utilizar o comando para que possamos observar os resultados.

#### **6.1.4 Achados em relação à intervenção sobre seus participantes**

Para avaliar os achados em relação à intervenção sobre os praticantes do estudo piloto investigou-se os seguintes fatores: i) Material instrucional e ii) Pré-

disposição para aprender. Tais fatores constituem as duas condições mínimas para ocorrência da Aprendizagem Significativa.

O Material instrucional foi produzido pelo autor com base na teoria da aprendizagem significativa de Ausubel, utilizando o princípio de diferenciação progressiva e reconciliação integrativa. Foi feita uma abordagem geral do tema com ideias mais inclusivas, para posteriormente, ir se diferenciando sobre os conteúdos menos inclusivos, pois de acordo com Ausubel, as ideias mais inclusivas estão no topo da estrutura cognitiva do aprendiz do que as ideias menos inclusivas. As atividades práticas tiveram como objetivo a realização da reconciliação dos conteúdos que foram abordados. Durante a intervenção, os alunos utilizavam o material e realizavam a sua leitura, muitas vezes tinham dúvidas pertinentes em relação aos temas que estavam sendo estudados e frequentemente apresentavam respostas criativas e elaboradas. Entretanto, durante as atividades práticas foi perceptível que o material precisava ser reestruturado em alguns pontos, pois os alunos encontraram dificuldades e se notou lacunas nas explicações do material. Assim que as dificuldades e lacunas da apostila iam surgindo novas explicações e exemplos eram apresentados e os alunos concluíam as atividades propostas, dando indícios de que estavam relacionando de forma substantiva (não literal) os conceitos abordados com os seus conhecimentos prévios. Dessa forma, conclui-se que as atividades continham relação com os conhecimentos prévios dos alunos e uma estrutura lógica, satisfazendo as condições necessárias para se considerado um material potencialmente significativo.

A pré-disposição dos estudantes para aprender foi avaliada a partir das observações do autor. Em um momento inicial da primeira intervenção do estudo piloto os alunos não se mostravam muito motivados, entretanto, com decorrer das explicações os alunos começaram a ficar interessados pelo assunto. Conforme se iniciou as atividades práticas o interesse dos alunos aumentou de forma considerável e a timidez que inicialmente era demonstrada, começa progressivamente a ficar de lado. Na metade da primeira intervenção até o final da mesma, os alunos já questionavam o autor de forma ativa e conseguiam fazer associações com exemplos do cotidiano. No início da segunda intervenção os alunos já demonstravam uma motivação inicial para aprender, ao ponto de que, pediram mais uma atividade para ser realizada, pois estes queriam continuar a programar.

O material potencialmente significativo e a pré-disposição para aprender são os principais fatores para a ocorrência de uma aprendizagem significativa. A necessidade de explicações adicionais nas quais não estavam presentes na apostila foi feita pelo autor que contornavam as falhas da primeira versão do material instrucional, entretanto, como um dos objetivos é a concepção de um material que possa ser potencialmente significativo, o Quadro 4 faz um levantamento das dúvidas, facilidades e dificuldades que os alunos obtiveram durante as intervenções.

Quadro 4 – Duvidas, Facilidades e Dificuldades.

| Dúvidas  | Facilidades                                   | Dificuldades  |
|--|---|---|
| Compreender em um primeiro momento qual o sentido de se trabalhar com os blocos lógicos; | Utilização de estruturas de condição;         | Compreender as conexões da <i>protoboard</i> ;          |
| Mudança nos parâmetros das estruturas de condição;                                       | Diferenciar operadores lógicos “e/ou”;        | Montar circuitos eletrônicos;                           |
| Aplicação da plataforma <i>Arduino</i> ;   | Raciocínio lógico na construção de programas; | Diferenciar variáveis inteiras das decimais;            |
| Funcionamento do sorteio dos blocos lógicos;   | Utilização do <i>Ardublock</i> .              | Atribuir um nome a uma variável para utiliza-la depois. |
| Qual tipo de variável utilizar;  | –   | –   |
| Como utilizar e montar circuitos na <i>protoboard</i> ;                                  | –   | –   |
| Como atribuir nome à variável.   | –   | –   |

Fonte: Autor (2019).

#### 6.1.4 Achados em relação a intervenção propriamente dita:

Com a realização do Estudo Piloto se tem como principal resultado as lacunas encontradas no material instrucional, pois a primeira versão da apostila por si só não poderia ser considerada como um material potencialmente significativo.

Ao aplicarmos a apostila em situação real de sala de aula identificamos lacunas em explicações e conteúdos apresentados em momentos que os alunos ainda não possuíam conhecimento prévio para darem significados às atividades propostas. Assim, foi necessária uma reestruturação da apostila e a inserção de novos tópicos e explicações. O Quadro 5 foi construído levando em conta o estudo piloto e o Quadro 4 como parâmetros para modificação da apostila.

Quadro 5 – Alterações na apostila.

| Versão preliminar  | Versão 1.1  |
|--|---|
| A seção dos blocos lógicos se encontrava antes da seção das estruturas de condição e operadores lógicos. | A seção dos blocos lógicos foi realocada para ficar depois das estruturas de condição e operadores lógicos.   |
| As atividades se encontravam logo após dos conteúdos necessários para a sua execução.                    | Todas as atividades foram colocadas no final de cada capítulo.  |
| O funcionamento da <i>protoboard</i> era sucinto e sem exemplos.   | Foi feita uma explicação mais aprofundada e adicionado exemplos.  |
| Não havia uma boa diferenciação entre variáveis inteiras e decimais.                                     | Foi explicitado e adicionado dicas as atividades envolvendo variáveis.  |
| Era apresentado apenas algumas funções do <i>Ardublock</i> e sem adição de atividades experimentais.     | Foram adicionadas as principais funções básicas do <i>Ardublock</i> com atividades experimentais, explicação de divisão de tensão, sensores, atuadores e projetos mais avançados. |

Fonte: Autor (2019).

Os objetivos da nova versão do material são: 1 – auxiliar na criação de subsunçores na estrutura cognitiva do aprendiz sobre a programação do

*Arduino/Ardublock* e da utilização de sensores/atuadores e 2 – servir como base para aqueles que desejarem trabalhar com projetos avançados que envolvam a aplicação *Ardublock*.

Analisando os objetivos específicos no qual se pretendia atingir com o estudo piloto foi possível notar que tais objetivos foram cumpridos, tendo em vista que: foi feito um levantamento das principais facilidades e dificuldades dos alunos, observaram-se as lacunas didáticas na primeira versão do material instrucional, analisou-se a pré-disposição dos alunos para aprender o tópico de estudo. Restando apenas as recomendações feitas pelo autor para aqueles que desejarem trabalhar com a programação em blocos no Ensino Fundamental.

A base para este último objetivo é o trabalho que Ribeiro (2017), no qual na conclusão de seu trabalho, apresenta 11 recomendações para aqueles que desejarem trabalhar com a plataforma *Arduino*. Partindo destas recomendações concebidas por Ribeiro, o autor utiliza, adapta e complementa tais orientações:

- I. Fazer uma associação geral dos conteúdos, com exemplos do cotidiano dos alunos, seguindo uma sequência lógica;
- II. Dar uma ênfase na montagem de circuitos eletrônicos, explicar o funcionamento de cada componente que está sendo utilizado;
- III. Aplicar atividades que relacionem os conteúdos aprendidos com a prática experimental e/ou da programação;
- IV. Introduzir inicialmente os conceitos da lógica de programação antes da linguagem de programação (em blocos);
- V. Diferenciar as semelhanças e as distinções entre códigos e comandos de programação;
- VI. Auxiliar quando os alunos apresentarem dificuldades sobre o tema de estudo, para que todos os alunos consigam concluir as atividades;
- VII. Aplicar as atividades relacionadas ao tema apenas após a apresentação dos todos os conteúdos necessários para sua execução, de modo que, os alunos possam ter os conhecimentos prévios para realizar tais atividades;
- VIII. Sempre que existir alguma dúvida, tente saná-la de forma coletiva pois, é possível que a dúvida de um aluno possa ser a mesma de seus colegas.

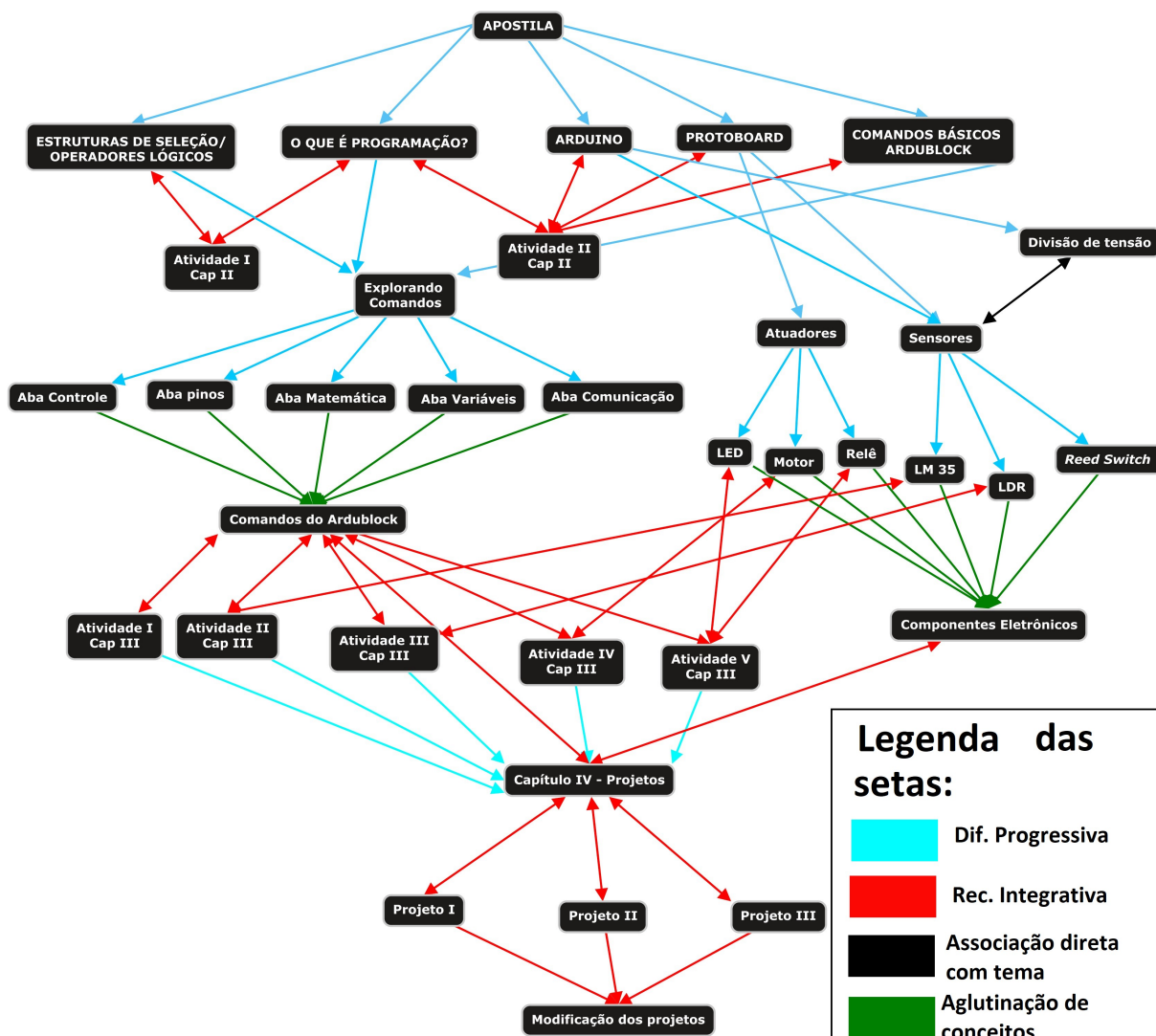
Em relação a apostila destaca-se que a mesma foi construída segundo o princípio programático da diferenciação progressiva, isto é, de uma abordagem que parte dos conceitos mais gerais para posteriormente serem diferenciados.

O Capítulo II descreve de forma clara e com exemplos do cotidiano, a ideia mais geral sobre a introdução a programação, pois de acordo com Ausubel, as ideias mais inclusivas, ficam no topo da estrutura cognitiva do aprendiz e a diferenciação deste conteúdo com conceitos e fatos menos inclusivos se tornam mais “claros” para o aprendiz e são diferenciados nos Capítulos subsequentes.

Porém, além da diferenciação progressiva, Ausubel fala sobre a reconciliação integrativa, isto é: “[...] *explorar, explicitamente, relações entre proposições e conceitos, chamar a atenção para diferenças e similaridades importantes e reconciliar inconsistências reais ou aparentes.*” (MOREIRA, 1982, p. 21). As atividades no final de cada Capítulo, juntamente com os projetos, têm como função fazer uma reconciliação integrativa com os conteúdos aprendidos.

Na Figura 21 apresenta-se um fluxograma com a diferenciação progressiva e reconciliação integrativa dos conteúdos:

Figura 21 – Diferenciação progressiva e reconciliação integrativa dos conteúdos.



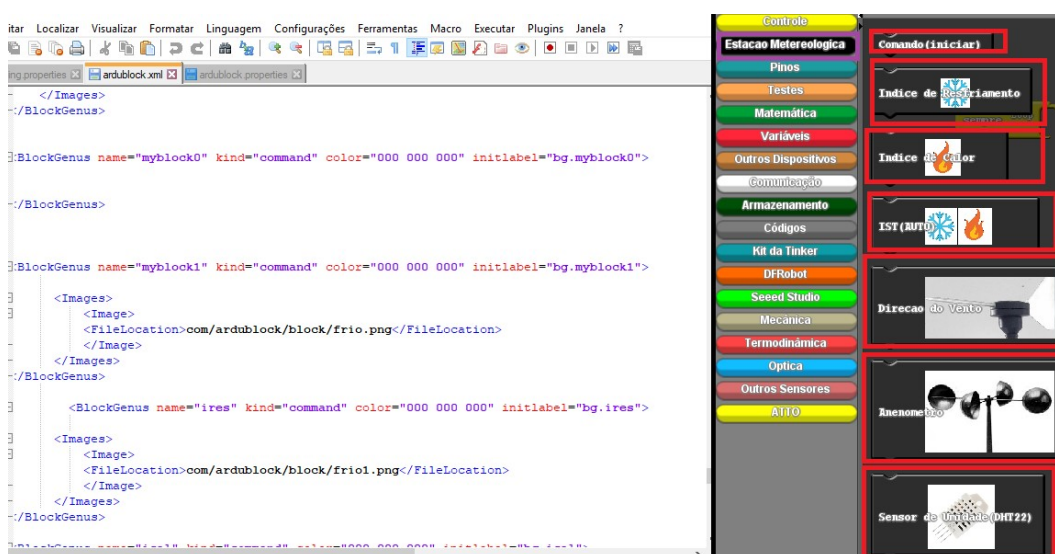
Fonte: Autor (2019).

Apostila tem como foco auxiliar na iniciação da programação e automação de pequenos projetos experimentais, sendo um recurso importante para propiciar conhecimentos prévios para a construção de projetos mais sofisticados. Nesse contexto, citamos a programação da miniestação meteorológica, introduzindo o ensino de programação na plataforma *Arduino* com o auxílio da aplicação *Ardublock*. Foram feitas modificações nesta aplicação, com o intuito de facilitar a programação da miniestação meteorológica, uma vez que, existem comandos que não estão presentes na versão do software disponibilizada pelo projeto TechDuino da Universidade Federal de Santa Catarina (UFSC), por exemplo: comandando `pulseIn` (utilizado para programar um anemômetro), comandos do DHT22. Podemos criar

blocos para equações muito grandes, como por exemplo: o cálculo do índice de resfriamento e índice de calor.

Através da programação em XML criamos novas abas e blocos, modificando arquivos que estão presentes dentro da raiz do *Ardublock*. Foi utilizado o *software* Notepad++ para edição destes arquivos. A Figura 22, podemos ver a esquerda a programação do código fonte em XML dos arquivos responsáveis por criar os blocos e a uma nova aba e o resultado de tais modificações a direita:

Figura 22 – Criação dos Blocos



Fonte: Autor (2019).

Entretanto, a criação dos blocos não traz consigo a programação, é necessário editar um arquivo em Java, e aqui, temos que programar em duas linguagens diferentes ao mesmo tempo, na Figura 23 podemos ver que os retângulos roxos são a programação em Java, enquanto o retângulo em vermelho é programação em C++ do *Arduino*.



Figura 23 – Programação do Código em Java

```

1 package com.ardublock.translator.block.Duinoedu;
2
3 import com.ardublock.translator.Translator;
4 import com.ardublock.translator.block.TranslatorBlock;
5 import com.ardublock.translator.block.exception.SocketNullException;
6 import com.ardublock.translator.block.exception.SubroutineNotDeclaredException;
7
8 public class ical extends TranslatorBlock{
9     public ical(Long blockId, Translator translator, String codePrefix, String codeSuffix, String label
10     {
11         super(blockId, translator, codePrefix, codeSuffix, label);
12     }
13     public String toCode() throws SocketNullException, SubroutineNotDeclaredException
14     {
15
16
17
18         String ret = "float tf = 32 + ((temp * 180) / 100); \n float sf = -42.379 + 2.04901523 * tf + 10.
19
20

```

Fonte: Autor (2019).

Existem diversos comandos em Java para colocarmos o código em diferentes partes, por exemplo: na inicialização de bibliotecas e atribuição de variáveis, *Void Setup()* e *Void Loop()*.

Entretanto, os últimos passos são: compilar o arquivo em java e transformá-lo no formato *.class* utilizando um compilador via *prompt de comando*. Após isso, é necessário editar este arquivo *.class* mudando o valor de sua leitura e colocar onde este arquivo ficará dentro da raiz do *Ardublock*. Posteriormente, é necessário colocar o arquivo no local onde foi especificado na edição do arquivo *.class*. O último passo, é programar em XML no arquivo de mapeamento dos códigos, ou seja, associar um determinado bloco com o arquivo que contém a programação. Após todos os passos serem realizados dentro do *Ardublock*, os novos comandos irão converter a linguagem do bloco para a linguagem do *Arduino*.

Os últimos passos da edição do *Ardublock* não foram adicionados neste trabalho, tendo em vista que: são muitos passos que devem ser realizados e será divulgado na *web* pós-conclusão deste TCC, um tutorial completo de como realizar modificações no *Ardublock*. As Figuras 23 e 24 mostram o mesmo código para o índice de resfriamento, entretanto a primeira é sem modificações e última é com as modificações.

Figura 23 – Índice de resfriamento (sem modificação)



Fonte: Autor (2019).

Figura 24 – Índice de resfriamento (com modificação)



Fonte: Autor (2019).

## 7 CONSIDERAÇÕES FINAIS

Durante o estudo piloto, foi observado pelo autor que a escola não tinha uma infraestrutura adequada, sendo que foi necessário levar os computadores para que ocorressem as intervenções. Ainda, por se tratar de uma escola na zona rural, em dias de chuva, não havia aula, pois, a estrada que leva até a escola fica inacessível. Entretanto, os alunos se mostraram motivados a aprender e o estudo propiciou a criação de um material instrucional que, tem como objetivo auxiliar na introdução à programação.

A necessidade da construção de um material instrucional sobre o ensino de programação no Ensino Fundamental partiu do princípio de que os alunos não possuíam conhecimentos prévios sobre programação. Deste modo, nos fundamentamos na teoria de aprendizagem significativa de Ausubel (2003), abordada anteriormente seção de referencial teórico, levando em consideração os conceitos de conhecimento prévio, diferenciação progressiva e reconciliação integrativa. Assim, foi concebida, implementada e avaliada uma apostila sobre o ensino de programação na plataforma *Arduino* com a utilização da ferramenta *Ardublock*. Pode-se observar que os alunos concluíram as atividades propostas, dando indícios de que estavam relacionando de forma substantiva (não literal) os conceitos abordados com os seus conhecimentos prévios. Dando indícios de que as atividades continham relação com os subsunçores dos alunos e uma estrutura lógica, satisfazendo, assim, as duas condições necessárias para a apostila ser considerada um material potencialmente significativo.

Durante a revisão da literatura, foi possível observar que não existem publicações utilizando o *Ardublock*, entretanto, durante este trabalho, esta ferramenta se mostrou com um alto potencial, uma vez que, é uma versão traduzida para o português, tem a maioria das funções de programação básica e existe a possibilidade de inserção de novos blocos de programação. Apesar de sua baixa utilização, apresenta uma vantagem sobre *Scrath for Arduino S4A*, pois permite que o usuário deixe o programa salvo na placa *Arduino*, possibilitando uma autonomia maior em relação à programação e possíveis projetos avançados como, por exemplo, a robótica.

Por fim cabe salientar, as modificações no *software* que foram realizadas pelo autor, não põem de lado a programação dos códigos, uma vez que, os alunos ainda

precisam realizar a sua programação. Tais modificações é apenas um facilitador para programações que são muito complexas para alunos do Ensino Fundamental, tendo em vista a aprendizagem significativa, os alunos não possuem os subsunçores necessários para realizar tais programações do zero.

## REFERÊNCIAS

- ALMEIDA, T. C.; DIAS, E. C.; JULIÃO, A. S. Um laboratório portátil de baixo custo: medição de  $g$  utilizando um pêndulo e a placa Raspberry Pi. **Caderno Brasileiro de Ensino de Física**, v. 34, n. 2, p. 590-602, 2017. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2017v34n2p590/34599>. Acesso em: 21 de ago. 2019.
- ALVARADO, M. S. A.; GALLEGOS, C. D. R. Mathematical model of a separately excited DC motor powered by a solar array using external starter resistances. **Latin-American Journal of Physics Education**, v. 8, n. 4, 2014. Disponível em: [http://www.lajpe.org/dec14/4307\\_Kamal\\_Chopra.pdf](http://www.lajpe.org/dec14/4307_Kamal_Chopra.pdf). Acesso em: 21 de ago. 2019.
- ALVES, E. G.; SILVA, A. F. Utilizando um led como fonte de energia. **A Física Na Escola**, v.9, n.1, 2008. Disponível em: <http://www.sbfisica.org.br/fne/Vol9/Num1/led.pdf>. Acesso em: 21 de ago. 2019.
- AMORIM, H. S.; DIAS, M. A.; SOARES, V. Sensores digitais de temperatura com tecnologia one-wire: Um exemplo de aplicação didática na área de condução térmica. **Revista Brasileira de Ensino de Física**, v. 37, n. 4, 2015. Disponível em: <http://www.scielo.br/pdf/rbef/v37n4/0102-4744-rbef-37-4-4310.pdf>. Acesso em: 21 de ago. 2019.
- AMRANI, D.; PARADIS, P. Use of computer-based data acquisition to teach physics laboratories: case study-simple harmonic motion. **Latin-American Journal of Physics Education**, v. 4, n. 3, p. 6, 2010. Disponível em: [http://www.lajpe.org/sep10/424\\_Djilali\\_Amrani.pdf](http://www.lajpe.org/sep10/424_Djilali_Amrani.pdf). Acesso em: 21 de ago. 2019.
- ANDA, L. T. C.; DÍAZ, M. H. R. Diseño, implementación e impacto de prototipos experimentales para mejorar la enseñanza de la ley de Biot-Savart en estudiantes de ingeniería. **Latin-American Journal of Physics Education**, v. 11, n. 2, 2017. Disponível em: [http://www.lajpe.org/jun17/2304\\_LTCD\\_2017.pdf](http://www.lajpe.org/jun17/2304_LTCD_2017.pdf). Acesso em: 21 de ago. 2019.
- ANJOS, J.R.; FREITAS, S.A.; ANDRADE NETO, A. S. Utilização do software *Scratch* para a aprendizagem de lançamentos de projéteis e conceito de gravidade no ensino fundamental. **ACTIO: Docência em Ciências**, v. 1, n. 1, p. 128-144, 2016. Disponível em: <https://periodicos.utfpr.edu.br/actio/article/download/5004/3224>. Acesso em: 21 de ago. 2019.

ARDUINO. **Definição do termo Arduino**. Disponível em: <https://playground.arduino.cc/Portugues/HomePage>. Acesso em: 11 out. 2018.

AUSUBEL, D. P. **Aquisição e Retenção de Conhecimentos: uma perspectiva cognitiva**. Lisboa: Plátano, 2003.

BALDO, D. A. *et al.* Aparato educacional para estudo da queda livre com análise do movimento. **Caderno Brasileiro de Ensino de Física**, v. 33, n. 3, p. 1064-1078, 2016. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2016v33n3p1064/33005>. Acesso em: 21 de ago. 2019.

BIAZUS, M. O.; ROSA, C. T. W. Abordagem de tópicos de mecânica quântica no ensino médio partindo da aproximação com o cotidiano. **Experiências em Ensino de Ciências**, v. 11, n. 3, p. 159-177, 2016. Disponível em: [http://if.ufmt.br/eenci/artigos/Artigo\\_ID330/v11\\_n3\\_a2016.pdf](http://if.ufmt.br/eenci/artigos/Artigo_ID330/v11_n3_a2016.pdf). Acesso em: 21 de ago. 2019.

BINI, E. M.; RESENDE, L. M. M.; SILVEIRA, R. M. C. F. Constante evolução da Tecnologia da Informação exige novo perfil dos jovens profissionais. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 2, n. 2, 2009. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/461/342>. Acesso em: 21 de ago. 2019.

BORROTO, H.; BECQUER, M.; AMAYA, O. S. Calzadilla. Evaluación de la Incertidumbre en la medición en el laboratorio introductorio de Física. **Latin-American Journal of Physics Education**, v. 11, n. 3, 2017. Disponível em: [http://www.lajpe.org/sep17/05\\_Calzadilla\\_3305.pdf](http://www.lajpe.org/sep17/05_Calzadilla_3305.pdf). Acesso em: 21 de ago. 2019.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular – BNCC 3ª versão**. Brasília, DF, 2017. Disponível em: [http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_sit\\_e.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_sit_e.pdf). Acesso em: 21 de ago. 2019

CALVALCANTE, M. A.; TEIXEIRA, A. C.; BALATON, M. Estudo das cores com Arduino *Scratch* e tracker. **A Física Na Escola**, v.14, n.1, 2016. Disponível em: <http://www1.fisica.org.br/fne/phocadownload/Vol14-Num1/fne-14-1-a071.pdf>. Acesso em: 21 de ago. 2019.

CARVALHO NETO, J. T.; APOLINÁRIO, F. R.; SOARES, A. Sistema *photogate* de seis canais analógicos para laboratórios didáticos de física. **Revista Brasileira de Ensino de Física**, v. 40, n. 1, 2017. Disponível em: <http://www.scielo.br/pdf/rbef/v40n1/1806-1117-rbef-40-01-e1504.pdf>. Acesso em: 21 de ago. 2019.

CARVALHO, L. R. M.; AMORIM, H. S. Observando as marés atmosféricas: Uma aplicação da placa Arduino com sensores de pressão barométrica e temperatura. **Revista Brasileira de Ensino de Física**, v. 36, n. 3, 2014. Disponível em: <http://www.scielo.br/pdf/rbef/v36n3/13.pdf>. Acesso em: 21 de ago. 2019.

CASTRO-PALACIO, J. C. *et al.* Using a smartphone acceleration sensor to study uniform and uniformly accelerated circular motions. **Revista Brasileira de Ensino de Física**, v. 36, n. 2, 2014. Disponível em: <http://www.scielo.br/pdf/rbef/v36n2/15.pdf>. Acesso em: 21 de ago. 2019.

CAVALCANTE, M. A.; RODRIGUES, C. E. M.; PONTES, L. A. Novas tecnologias no estudo de ondas sonoras. **Caderno Brasileiro de Ensino de Física**, v. 30, n. 3, p. 579-613, 2013. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2013v30n3p579/25604>. Acesso em: 21 de ago. 2019.

CAVALCANTE, M. A.; RODRIGUES, T. T. T.; BUENO, D. A. Controle Remoto: observando códigos com o Arduino (parte 2 de 2). **Caderno Brasileiro de Ensino de Física**, v. 31, n. 3, p. 614-641, 2014. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2014v31n3p614/27970>. Acesso em: 21 de ago. 2019.

CAVALCANTE, M. A.; RODRIGUES, T. T. T.; BUENO, D. A. Controle Remoto: princípio de funcionamento (parte 1 de 2). **Caderno Brasileiro de Ensino de Física**, v. 30, n. 3, p. 554-565, 2013. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2013v30n3p554/25602>. Acesso em: 21 de ago. 2019.

CAVALCANTE, M. A.; TAVOLARO, C. R. C.; MOLISANI, E. Física com Arduino para iniciantes. **Revista Brasileira de Ensino de Física**, v. 33, n. 4, 2011. Disponível em: <http://www.sbfisica.org.br/rbef/pdf/334503.pdf>. Acesso em: 21 de ago. 2019.

COLUCI, V. R. *et al.* Ilustração de incertezas em medidas utilizando experimentos de queda livre **Revista Brasileira de Ensino de Física**, v. 35, n. 2, 2013. Disponível em: <http://www.scielo.br/pdf/rbef/v35n2/26.pdf>. Acesso em: 21 de ago. 2019.

CORDOVA, H. P. *et al.* Audietermômetro: um termômetro para a inclusão de estudantes com deficiência visual. **Revista Brasileira de Ensino de Física**, v. 40, n. 2, 2018. Disponível em: <http://www.scielo.br/pdf/rbef/v40n2/1806-1117-rbef-40-02-e2505.pdf>. Acesso em: 21 de ago. 2019.

CORDOVA, H.; TORT, A. C. Medida de g com a placa Arduino em um experimento simples de queda livre. **Revista Brasileira de Ensino de Física**, v. 38, n. 2, 2016. Disponível em: <http://www.scielo.br/pdf/rbef/v38n2/1806-1117-rbef-38-02-e2308.pdf>. Acesso em: 21 de ago. 2019.

DAMIANI, M. F. *et al.* Discutindo pesquisas do tipo intervenção pedagógica. **Cadernos de Educação**, n. 45, p. 57- 67, 2013. Disponível em: <https://periodicos.ufpel.edu.br/ojs2/index.php/caduc/article/download/3822/3074>. Acesso: 21 de ago. 2019.

DIONISIO, G.; SPALDING, L. E. S. Visualização da forma de onda e conteúdo harmônico da corrente elétrica alternada em eletrodomésticos. **Revista Brasileira de Ensino de Física**, v. 39, n. 1, 2017. Disponível em: <http://www.scielo.br/pdf/rbef/v39n1/1806-1117-rbef-39-01-e1501.pdf>. Acesso em: 21 de ago. 2019

DWORAKOWSKI, L. A. *et al.* Uso da plataforma Arduino e do software PLX-DAQ para construção de gráficos de movimento em tempo real. **Revista Brasileira de Ensino de Física**, v. 38, n. 3, 2016. Disponível em: <http://www.scielo.br/pdf/rbef/v38n3/1806-1117-rbef-38-03-e3503.pdf>. Acesso em: 21 de ago. 2019.

ESPINDOLA, P. R. *et al.* Low cost MFC control unit using microcontoller. **Revista Brasileira de Física Tecnológica Aplicada**, v. 4, n. 2, 2017. Disponível em: <https://periodicos.utfpr.edu.br/rbfta/article/download/5581/4620>. Acesso em: 21 de ago. 2019.



ESPINOZA, M. Q; ZAVALA, G. El uso de calculadoras con sensores en el aprendizaje de circuitos eléctricos. **Latin-American Journal of Physics Education**, v. 8, n. 4, 2014. Disponível em: [http://www.lajpe.org/dec14/4507\\_Zavala.pdf](http://www.lajpe.org/dec14/4507_Zavala.pdf). Acesso em: 21 de ago. 2019.

FERNANDES, M. B.; HARTMANN, A. M.; DORNELES, P. F. T. A física no ensino médio integrado: uma sequência didática sobre eletricidade com aplicação do Arduino. **Revista Brasileira de Física Tecnológica Aplicada**, v. 1, n. 2, 2014. Disponível em: <https://periodicos.utfpr.edu.br/rbfta/article/download/2095/1814>. Acesso em: 21 de ago. 2019.

GERMÁN, M. *et al.* Estudio de la caída libre utilizando diferentes técnicas experimentales. **Latin-American Journal of Physics Education**, v. 12, 2018. Disponível em: [http://www.lajpe.org/mar18/12\\_1\\_02.pdf](http://www.lajpe.org/mar18/12_1_02.pdf). Acesso em: 21 de ago. 2019.

GONZÁLEZ, V. V.; MACÍAS, C. A.; ZAPATA, S. H. Obtención de las curvas de saturación de alcohol etílico. **Latin-American Journal of Physics Education**, v. 4, n. 1, p. 9, 2010. Disponível em: [http://www.lajpe.org/LAJPE\\_AAPT/09\\_Valentin\\_Valdes.pdf](http://www.lajpe.org/LAJPE_AAPT/09_Valentin_Valdes.pdf). Acesso em: 21 de ago. 2019.

GOYA, A.; LABURÚ, C. E. Uma atividade experimental de Física por meio de investigação multimodal representacional. **Experiências em Ensino de Ciências**, v. 9, n. 2, p. 32-44, 2014. Disponível em: [http://if.ufmt.br/eenci/artigos/Artigo\\_ID241/v9\\_n2\\_a2014.pdf](http://if.ufmt.br/eenci/artigos/Artigo_ID241/v9_n2_a2014.pdf). Acesso em: 21 de ago. 2019.

GRINGS, E. T. O.; CABALLERO, C.; MOREIRA, M. A. Uma proposta didática para abordar o conceito de temperatura a partir de situações, à luz da teoria dos campos conceituais de Vergnaud. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 1, n. 1, 2008. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/221/213>. Acesso em: 21 de ago. 2019.

GUADAGNINI, P. H.; ROCHA, F. S.; BARLETTE, V. E. Projeto de um sensor eletrônico baseado em extensometria para medição de força. **Latin-American Journal of Physics Education**, v. 5, n. 4, 2011. Disponível em: [http://www.lajpe.org/dec11/LAJPE\\_590\\_Paulo\\_Guadagnini\\_preprint\\_corr\\_f.pdf](http://www.lajpe.org/dec11/LAJPE_590_Paulo_Guadagnini_preprint_corr_f.pdf). Acesso em: 21 de ago. 2019.

GUTIERRE, H. G. *et al.* Sistema fotodetector económico para utilização em laboratórios de ensino e pesquisa. **Revista Brasileira de Ensino de Física**, v. 39, n. 3, 2017. Disponível em: <http://www.scielo.br/pdf/rbef/v39n3/1806-1117-rbef-39-03-e3501.pdf>. Acesso em: 21 de ago. 2019.

IANNELLI, L.; GIL, S. Acondicionamiento térmico de aire usando energía geotérmica-ondas de calor. **Latin-American Journal of Physics Education**, v. 6, n. 1, 2012. Disponível em: [http://www.lajpe.org/mar12/18\\_LAJPE\\_617\\_Salvador\\_Gil\\_preprint\\_corr\\_f.pdf](http://www.lajpe.org/mar12/18_LAJPE_617_Salvador_Gil_preprint_corr_f.pdf). Acesso em: 21 de ago. 2019.

JESUS, V. L. B.; SASAKI, D. G. G. Uma visão diferenciada sobre o ensino de forças impulsivas usando um smartphone. **Revista Brasileira de Ensino de Física**, v. 38, n. 1, 2016. Disponível em: <http://www.scielo.br/pdf/rbef/v38n1/1806-9126-rbef-38-01-1303.pdf>. Acesso em: 21 de ago. 2019.

KELLY, G.; ROCHA, D.; GERMANO, R. Espectroscopia para o Ensino Médio utilizando a placa Arduino. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 10, n. 2, 2017. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/3997/pdf>. Acesso em: 21 de ago. 2019.

LEMUS, J.; SERRA, R.; COLLAZO, R. Los diodos láser en la Enseñanza de la Física en Ingeniería. **Latin-American Journal of Physics Education**, v. 5, n. 4, p. 800, 2011. Disponível em: [http://www.lajpe.org/dec11/LAJPE\\_594\\_Jose\\_Lemus\\_preprint\\_corr\\_f.pdf](http://www.lajpe.org/dec11/LAJPE_594_Jose_Lemus_preprint_corr_f.pdf). Acesso em: 21 de ago. 2019.

LORETO, E. L. S.; SARTORI, P. H. S. Simulação da visão das cores: decodificando a transdução quântica-elétrica. **Caderno Brasileiro de Ensino de Física**, v. 25, n. 2, p. 266-286, 2008. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2008v25n2p266/5635>. Acesso em: 21 de ago. 2019.

MACÊDO, J. A.; PEDROSO, L. S.; COSTA, G. A. Aprimorando e validando um fotogate de baixo custo. **Revista Brasileira de Ensino de Física**, v. 40, n. 4, 2018. Disponível em: <http://www.scielo.br/pdf/rbef/v40n4/1806-9126-RBEF-40-4-e5403.pdf>. Acesso em: 21 de ago. 2019.

MARTINS, J. E. M. P.; VIANNA, A. C. G. Teaching general concepts about sensors and transfer functions with a voltage divider. **Latin-American Journal of Physics Education**, v. 5, n. 4, 2011. Disponível em: [http://www.lajpe.org/dec11/LAJPE\\_587\\_Joao\\_Perea\\_preprint\\_corr\\_f.pdf](http://www.lajpe.org/dec11/LAJPE_587_Joao_Perea_preprint_corr_f.pdf). Acesso em: 21 de ago. 2019.

MATHIAS, L. C.; SILVA, P. R. C.; SILVA, O. H. M. Um instrumento alternativo ao estudo de pilhas recarregáveis via Arduino. **Caderno Brasileiro de Ensino de Física**, v. 34, n. 1, p. 310-323, 2017. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2017v34n1p310/33957>. Acesso em: 21 de ago. 2019.

MONTEIRO, D. C.; CELESTE, A. T. B. Construção de um termômetro para fins didáticos. **A Física Na Escola**, v.13, n.1, 2012. Disponível em: <http://www1.fisica.org.br/fne/phocadownload/Vol13-Num1/a101.pdf>. Acesso em: 21 de ago. 2019.

MONTEIRO, E. L.; CAMARGO FILHO, P. S.; GRESCZYSCZYN, M. C. C. Atividade experimental como recurso para interação de alunos com transtornos específicos de aprendizagem em Física Moderna e Contemporânea. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 10, n. 1, 2017. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/5715/pdf>. Acesso em: 21 de ago. 2019.

MORALES, C. A. C. Construcción de un prototipo para experimentos de mecánica. **Latin-American Journal of Physics Education**, v. 4, n. 1, p. 1, 2010. Disponível em: [http://www.lajpe.org/LAJPE\\_AAPT/01\\_Carlos\\_Collazos.pdf](http://www.lajpe.org/LAJPE_AAPT/01_Carlos_Collazos.pdf). Acesso em: 21 de ago. 2019.

MOREIRA, M. A. **Aprendizagem significativa: a teoria de David Ausubel**. São Paulo: Moraes, 1982.

MÜTZENBERG, L. A.; VEIT, E. A.; SILVEIRA, F. L. Avaliação dos trabalhos trimestrais com base na opinião dos alunos. **Experiências em Ensino de Ciências**. v. 2, n. 3, p. 01-16, 2007. Disponível em: [https://www.if.ufrgs.br/~lang/Textos/2007\\_2\\_3\\_37.pdf](https://www.if.ufrgs.br/~lang/Textos/2007_2_3_37.pdf). Acesso em: 21 de ago. 2019.

MÜTZENBERG, L. A.; VEIT, E. A.; SILVEIRA, F. L. Trabalhos trimestrais: uma proposta de pequenos projetos de pesquisa no ensino da física. **Experiências em Ensino de Ciências**, v. 2, n. 2, p. 11-22, 2007. Disponível em: [http://if.ufmt.br/eenci/artigos/Artigo\\_ID38/pdf/2007\\_2\\_2\\_38.pdf](http://if.ufmt.br/eenci/artigos/Artigo_ID38/pdf/2007_2_2_38.pdf). Acesso em: 21 de ago. 2019.

OLIVEIRA JÚNIOR, R.L.; ALVES J.R.M.; BARBOSA, V. H. Aquecimento e resfriamento da água, aproximados à forma real. **Caderno Brasileiro de Ensino de Física**, v. 33, n. 1, p. 306-319, 2016. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2016v33n1p306/31591>. Acesso em: 21 de ago. 2019.

OLIVEIRA, I. N. *et al.* As mudanças ocorridas nos programas de ensino da física, os laboratórios didáticos de física e a inclusão de novas tecnologias no desenvolvimento dos experimentos remotos. **Experiências em Ensino de Ciências**, v. 11, n. 3, p. 51-68, 2016. Disponível em: [http://if.ufmt.br/eenci/artigos/Artigo\\_ID323/v11\\_n3\\_a2016.pdf](http://if.ufmt.br/eenci/artigos/Artigo_ID323/v11_n3_a2016.pdf). Acesso em: 21 de ago. 2019.

OLIVEIRA, I. N. *et al.* Automatização de laboratórios didáticos de física geral - construção de uma maquete experimental automatizada para a determinação da constante de Planck. **Experiências em Ensino de Ciências**, v. 13, n. 1, p. 131-153, 2018. Disponível em: [http://if.ufmt.br/eenci/artigos/Artigo\\_ID462/v13\\_n1\\_a2018.pdf](http://if.ufmt.br/eenci/artigos/Artigo_ID462/v13_n1_a2018.pdf). Acesso em: 21 de ago. 2019.

PEDROSO, L. S.; PIMENTA NETO, F.; ARAÚJO, M. S. T. Investigação sobre o funcionamento de um termômetro digital de baixo custo. **Revista Brasileira de Física Tecnológica Aplicada**, v. 1, n. 2, 2014. Disponível em: <https://periodicos.utfpr.edu.br/rbfta/article/download/1879/1813>. Acesso em: 21 de ago. 2019.

PENEREIRO, J. C. Medidas das eficiências térmicas em aquecedores solares: uma alternativa complementar para o ensino de conceitos de física para estudantes das engenharias. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 9, n. 1, 2016. Disponível em: <https://revistas.utfpr.edu.br/rbect/article/download/1598/2960>. Acesso em: 21 de ago. 2019.

PEREIRA, A. M.; SANTOS, A. C. F.; AMORIM, H. S. Estatística de contagem com a plataforma Arduino. **Revista Brasileira de Ensino de Física**, v. 38, n. 4, 2016. Disponível em: <http://www.scielo.br/pdf/rbef/v38n4/1806-1117-rbef-38-04-e4501.pdf>. Acesso em: 21 de ago. 2019.

PIRES, C. F. J. S.; FERRARI, P. C.; QUEIROZ, J. R. O. A tecnologia do motor elétrico para o ensino de Eletromagnetismo numa abordagem problematizadora. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 6, n. 3, 2014. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/1525/1141>. Acesso em: 21 de ago. 2019.

RIBEIRO, J.D. **Explorando as possibilidades de inserção da plataforma Arduino no Ensino de Ciências da Educação Básica**. 2015. 168 f. Dissertação de Mestrado. Curso de Pós-graduação em Ensino de Ciências, Universidade Federal do Pampa, Bagé, 2017. Disponível em: <http://dspace.unipampa.edu.br:8080/jspui/bitstream/riu/3044/1/Disserta%20a7%20a3o%20Janu%20a1rio%20Ribeiro%202017.pdf> Acesso em: 21 de ago. 2019.

ROCHA, F. S.; GUADAGNINI, P. H. Projeto de um sensor de pressão manométrica para ensino de física em tempo real. **Caderno Brasileiro de Ensino de Física**, v. 31, n. 1, p. 124-148, 2014. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2014v31n1p124/26448>. Acesso em: 21 de ago. 2019.

ROCHA, F. S.; GUADAGNINI, P. H. Sensor sonar de movimento para ensino de Física experimental. **Latin-American Journal of Physics Education**, v. 4, n. 2, p. 7, 2010. Disponível em: [http://www.lajpe.org/may10/07\\_Fabio\\_Saraiva.pdf](http://www.lajpe.org/may10/07_Fabio_Saraiva.pdf). Acesso em: 21 de ago. 2019.

ROCHA, F. S.; GUADAGNINI, P. H.; LUCCHESI, M. M. Projeto de um calorímetro de relaxação para ensino de Física. **Caderno Brasileiro de Ensino de Física**, v. 34, n. 1, p. 278-297, 2017. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2017v34n1p278/33987>. Acesso em: 21 de ago. 2019.

ROCHA, F. S.; MARANGHELLO, G. F.; LUCCHESI, M. M. Acelerômetro eletrônico e a placa Arduino para ensino de física em tempo real. **Caderno Brasileiro de Ensino de Física**, v. 31, n. 1, p. 98-123, 2014. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2014v31n1p98/26427>. Acesso em: 21 de ago. 2019

ROCHA, F. S.; MARRANGHELLO, G. F. Propriedades de um acelerômetro eletrônico e possibilidades de uso no ensino de mecânica. **Latin-American Journal of Physics Education**, v. 7, n. 1, 2013. Disponível em: [http://www.lajpe.org/march13/6\\_LAJPE\\_739\\_Fabio\\_Saraiva\\_preprint\\_corr\\_f.pdf](http://www.lajpe.org/march13/6_LAJPE_739_Fabio_Saraiva_preprint_corr_f.pdf). Acesso em: 21 de ago. 2019.

ROSA, C. T. W. *et al.* Experimento de condução térmica com e sem uso de sensores e Arduino. **Caderno Brasileiro de Ensino de Física**, v. 33, n. 1, p. 292-305, 2016. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2016v33n1p292/31590>. Acesso em: 21 de ago. 2019.

ROSENBERG, J.; CUFF, K. Low-cost sensing to teach energy for everyone. **Latin-American Journal of Physics Education**, v. 1, n. 39, 2011. Disponível em: [http://www.lajpe.org/icpe2011/7\\_Joel\\_Rosenberg.pdf](http://www.lajpe.org/icpe2011/7_Joel_Rosenberg.pdf). Acesso em: 21 de ago. 2019.

SANTOS, A. A. M.; AMORIM, H. S.; DEREZYNSKI, C. P. Investigação do fenômeno ilha de calor urbana através da utilização da placa Arduino e de um sítio oficial de meteorologia. **Revista Brasileira de Ensino de Física**, v. 39, n. 1, 2017. Disponível em: <http://www.scielo.br/pdf/rbef/v39n1/1806-1117-rbef-39-01-e1505.pdf>. Acesso em: 21 de ago. 2019.

SAPIÉN, P. V.; CHAVARRÍA, G. R. Transferencia de calor en un concentrador solar. **Latin-American Journal of Physics Education**, v. 4, n. 1, p. 12, 2010. Disponível em: [http://www.lajpe.org/LAJPE\\_AAPT/12\\_Pamela\\_Villamil.pdf](http://www.lajpe.org/LAJPE_AAPT/12_Pamela_Villamil.pdf). Acesso em: 21 de ago. 2019.

SILVA, A. C. Uma Possível Associação entre Ciência e Tecnologia no Ensino de Ciências: o exemplo do telefone. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 6, n. 3, 2014. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/1426/1149>. Acesso em: 21 de ago. 2019.

SILVA, R. B. *et al.* Estações meteorológicas de código aberto: um projeto de pesquisa e desenvolvimento tecnológico. **Revista Brasileira de Ensino de Física**, v. 37, n. 1., 2015. Disponível em: <http://www.scielo.br/pdf/rbef/v37n1/0102-4744-rbef-37-01-1505.pdf>. Acesso em: 21 de ago. 2019.

SILVA, S. L. *et al.* Avaliação do módulo da aceleração da gravidade com Arduino. **Caderno Brasileiro de Ensino de Física**, v. 33, n. 2, p. 619-640, 2016. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2016v33n2p619/32333>. Acesso em: 21 de ago. 2019.

SILVEIRA JÚNIOR, C. R.; COELHO, J. D.; SANTOS, L. S. Robótica nas aulas de matemática do ensino médio: uma proposta educacional e de baixo custo. **Experiências em Ensino de Ciências**, v. 12, n. 5, p. 82-104, 2017. Disponível em: <https://periodicos.ufpel.edu.br/ojs2/index.php/caduc/article/download/3822/3074>. Acesso em: 21 de ago. 2019.

SILVEIRA, S.; GIRARDI, M. Desenvolvimento de um kit experimental com Arduino para o ensino de Física Moderna no Ensino Médio. **Revista Brasileira de Ensino de Física**, v. 39, n. 4, 2017. Disponível em: <http://www.scielo.br/pdf/rbef/v39n4/1806-1117-rbef-39-04-e4502.pdf>. Acesso em: 21 de ago. 2019.

SOARES, R. R.; BORGES, P. F. O plano inclinado de Galileu: uma medida manual e uma medida com aquisição automática de dados. **Revista Brasileira de Ensino de Física**, v. 32, n. 2, 2010. Disponível em: <http://www.scielo.br/pdf/rbef/v32n2/v32n2a12.pdf>. Acesso em: 21 de ago. 2019.

SOUZA, A. R. *et al.* A placa Arduino: uma opção de baixo custo para experiências de física assistidas pelo PC. **Revista Brasileira de Ensino de Física**, v. 33, n. 1, 2011. Disponível em: <http://sbfisica.org.br/rbef/pdf/331702.pdf>. Acesso em: 21 de ago. 2019.

SOUZA, E. C.; YONEZAWA, W. M. Programação no ensino de matemática utilizando Processing 2: Um estudo das relações formalizadas por alunos do ensino fundamental com baixo rendimento em matemática. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 11, n. 1, 2018. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/5766/pdf>. Acesso em: 21 de ago. 2019.

SZMOSKI, R. M. *et al.* Desenvolvimento de um aparato experimental de baixo custo para o estudo de objetos em queda: análise do movimento de magnetos em tubos verticalmente orientados. **Revista Brasileira de Ensino de Física**, v. 40, n. 1, 2017. Disponível em: <http://www.scielo.br/pdf/rbef/v40n1/1806-1117-rbef-40-01-e1505.pdf>. Acesso em: 21 de ago. 2019.

TOMAZIO, N. B.; ROMERO, A. L. S.; MENDONÇA, C. R. Desenvolvimento de um obturador de feixe óptico utilizando um disco rígido de computador. **Revista Brasileira de Ensino de Física**, v. 40, n. 1, 2017. Disponível em: <http://www.scielo.br/pdf/rbef/v40n1/1806-1117-rbef-40-01-e1307.pdf>. Acesso em: 21 de ago. 2019.

TRENTIN, M. A. S. *et al.* Robótica educativa livre no ensino de Física: da construção do robô à elaboração da proposta didática de orientação metacognitiva. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 8, n. 3, 2015. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/download/1950/2190>. Acesso em: 21 de ago. 2019.

USIBE, B. E.; ADIAKPAN, E. S.; OBU, J. A. Design, construction and testing of a vibrometer. **Latin-American Journal of Physics Education**, v. 7, n. 2, 2013. Disponível em: [http://www.lajpe.org/jun13/LAJPE\\_748\\_Brian\\_Usibe\\_preprint.pdf](http://www.lajpe.org/jun13/LAJPE_748_Brian_Usibe_preprint.pdf). Acesso em: 21 de ago. 2019.

VARANIS, M. *et al.* Instrumentation for mechanical vibrations analysis in the time domain and frequency domain using the Arduino platform. **Revista Brasileira de Ensino de Física**, v. 38, n. 1, 2016. Disponível em: <http://www.scielo.br/pdf/rbef/v38n1/1806-9126-rbef-38-01-1301.pdf>. Acesso em: 21 de ago. 2019.



VARANIS, Marcus; SILVA, Anderson Langone; MERELES, Arthur Guilherme. On mechanical vibration analysis of a multi degree of freedom system based on arduino and MEMS accelerometers. **Revista Brasileira de Ensino de Física**, v. 40, n. 1, 2018. Disponível em: <http://www.scielo.br/pdf/rbef/v40n1/1806-1117-rbef-40-01-e1304.pdf>. Acesso em: 21 de ago. 2019.

VEIT, E. A.; SILVA, L. F. Uma experiência didática com aquisição automática de dados no laboratório de física do ensino médio. **Experiências em Ensino de Ciências**, v. 1, n. 3, p. 18-32, 2006. Disponível em: [http://if.ufmt.br/eenci/artigos/Artigo\\_ID12/pdf/2006\\_1\\_3\\_12.pdf](http://if.ufmt.br/eenci/artigos/Artigo_ID12/pdf/2006_1_3_12.pdf). Acesso em: 21 de ago. 2019.

VICENTINI, A. *et al.* Instrumentação para o ensino de física moderna e sua inserção em escolas de ensino médio—relato de experiência. **Experiências em Ensino de Ciências**, v. 6, n. 3, p. 38-44, 2011. Disponível em: [http://if.ufmt.br/eenci/artigos/Artigo\\_ID157/v6\\_n3\\_a2011.pdf](http://if.ufmt.br/eenci/artigos/Artigo_ID157/v6_n3_a2011.pdf). Acesso em: 21 de ago. 2019.

VIEIRA, Leonardo.P; AGUIAR, Carlos Eduardo. Mecânica do acelerômetro de smartphones e tablets. **A Física Na Escola**, v.14, n.1, 2016. Disponível em: <http://www.sbfisica.org.br/fne/Vol14/Num1/fne-14-1-a03.pdf>. Acesso em: 21 de ago. 2019.

VILAR, A. B. *et al.* Medição de temperatura: O saber comum ignorado nas aulas experimentais. **Revista Brasileira de Ensino de Física**, v. 37, n. 2, 2015. Disponível em: <http://www.scielo.br/pdf/rbef/v37n2/0102-4744-rbef-37-02-2507.pdf>. Acesso em: 21 de ago. 2019.

VISCOVINI, R. C. *et al.* Maquete didática de um sistema trifásico de corrente alternada com Arduino: ensinando sobre a rede elétrica. **Caderno Brasileiro de Ensino de Física**, v. 32, n. 3, p. 856-869, 2015. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/download/2175-7941.2015v32n3p856/30640>. Acesso em: 21 de ago. 2019.

XIMENES, R. L. *et al.* Development and assembly of Radio Frequency Module using components with SMD technology for educational purposes. **Latin-American Journal of Physics Education**, v. 11, n. 1, 2017. Disponível em: [http://www.lajpe.org/mar17/1304\\_Ximenes\\_2017.pdf](http://www.lajpe.org/mar17/1304_Ximenes_2017.pdf). Acesso em: 21 de ago. 2019.

## APÊNDICE A – Apostila Versão 1.1

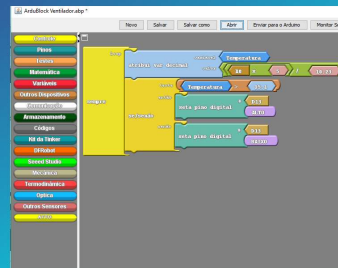


**APOSTILA DE INICIAÇÃO À  
PROGRAMAÇÃO EM BLOCOS COM A  
PLATAFORMA ARDUINO-  
ARDUBLOCK**

**Autor:**  
**Ricardo Gomes Lopes**

**Orientador:**  
**Pedro Fernando Teixeira  
Dorneles**

**Coorientador:**  
**Januário Dias Ribeiro**



## SUMÁRIO

|   |     |
|---|-----|
| CAPÍTULO I – INTRODUÇÃO.....                          | 77  |
| CAPÍTULO II – INICIAÇÃO À PROGRAMAÇÃO .....           | 78  |
| 2.1 O que é programação?.....                         | 78  |
| 2.2 Estruturas de seleção e operadores lógicos.....   | 80  |
| 2.2.1 Estruturas de Condição “Se” e “Senão” .....     | 81  |
| 2.2.2 Operadores Lógicos “e” e “ou” .....             | 83  |
| 2.2.3 Blocos Lógicos.....                             | 84  |
| 2.2 O que é o Arduino? .....                          | 86  |
| 2.3 <i>Protoboard</i> .....                           | 88  |
| 2.4 O que é o Ardublock e qual comandos básicos?..... | 90  |
| 2.5 Guia de Atividades do Capítulo II .....           | 92  |
| 2.6.1 Bingo Lógico .....                              | 93  |
| 2.6.2Blink (Acender e apagar um LED) .....            | 94  |
| 2.6.2.1 Montagem do Circuito.....                     | 94  |
| 2.6.2.2 Programação .....                             | 97  |
| Capítulo III - Explorando Recursos.....               | 98  |
| 3.1 Explorando mais comandos do Ardublock.....        | 98  |
| 3.1.1 Aba Controle.....                               | 98  |
| 3.1.1.1 Comandos para iniciar um programa.....        | 98  |
| 3.1.1.2 Comandos de tempo .....                       | 99  |
| 3.1.1.3 Estruturas de seleção e Aba de Testes .....   | 100 |
| 3.1.2 Aba Pinos .....                                 | 101 |
| 3.1.4 Aba Matemática.....                             | 102 |
| 3.1.5 Aba Variáveis.....                              | 102 |
| 3.1.7 Aba Comunicação .....                           | 103 |
| 3.2 Divisor de tensão .....                           | 104 |
| 3.3 Sensores .....                                    | 107 |
| 3.3.1 LM 35.....                                      | 108 |
| 3.3.2 LDR .....                                       | 109 |
| 3.3.3 Reed Switch .....                               | 109 |
| 3.4 Atuadores .....                                   | 110 |
| 3.4.1 LED.....  | 110 |
| 3.4.2 Motor .....                                     | 111 |
| 3.4.3 Relê .....                                      | 111 |
| 3.5 Atividades do Capítulo III.....                   | 113 |
| 3.5.1 Construindo um Cronômetro .....                 | 113 |
| 3.5.2 Medindo a temperatura.....                      | 113 |
| 3.5.3 Sensor de Luminosidade .....                    | 114 |
| 3.5.4 Ligando um Motor.....                           | 114 |

|   |     |
|---|-----|
| 3.5.5 Utilizando uma relê para ligar um LED .....           | 115 |
| CAPITULO IV- PROJETOS .....                                 | 116 |
| 4.1 Projeto I – Controlando uma Lâmpada com o Arduino ..... | 116 |
| 4.1.1 Montagem do Circuito do Projeto I .....               | 116 |
| 4.2 Projeto II – Ventilador Inteligente .....               | 116 |
| 4.2.1 Montagem do Circuito do Projeto II .....              | 102 |
| 4.3 Projeto III – Sinaleira inteligente .....               | 117 |
| 4.3.1 Montagem do Circuito do projeto III .....             | 117 |
| 4.4 Modificação dos Projetos .....                          | 118 |
| SOLUÇÕES: .....   | 119 |
| 5 REFERÊNCIAS .....   | 125 |
| GUIA DE INSTALAÇÃO ARDUBLOCK .....                          | 126 |
| CARTERLAS BINGO LÓGICO .....                                | 129 |

## Capítulo I – Introdução

Esta apostila é fruto do meu trabalho de conclusão de curso, sendo o tema deste o ensino do divisor de tensão através de experimentos utilizando o Arduino e a extensão Ardublock para programação. Deste material ainda constará um guia, no ANEXO A, de como se instalar o Arduino IDE e Ardublock nos sistemas operacionais Windows e Linux. A apostila contém três capítulos referentes às aulas, sendo estes: *Capítulo II- Iniciação programação*, *Capítulo III – Explorando recursos* e *Capítulo IV- Projetos*. O primeiro item desta lista tem como objetivo dar um panorama geral dos conceitos que serão abordados, tais como: o que é programação e o que é um algoritmo; estruturas de condição, operadores lógicos; funcionamento da *protoboard*; como funciona o Ardublock e um primeiro contato com circuitos elétricos e a programação em blocos. No Capítulo III, será explorado de forma bastante ampla e com exemplos as principais abas do Ardublock, os atuadores, relê, motor e LED, os sensores LM35, LDR, e *reed switch* e o divisor de tensão e o Capítulo IV, com conhecimentos adquiridos, os alunos deverão realizar projetos com sensores e atuadores.

## Capítulo II – Iniciação à programação

### 2.1 O que é programação?

Uma das primeiras ideias que precisamos ter em nossas mentes é que o computador não entende a linguagem humana, mas sim, a linguagem binária (0 e 1). Assim, podemos dizer que a programação é a transposição de uma linguagem humana para uma linguagem computacional. Para programarmos precisamos de um *hardware* que é um computador e um *software* que é um programa, conforme ilustrado na Figura 1.

Figura 1 – Linguagem de Programação.

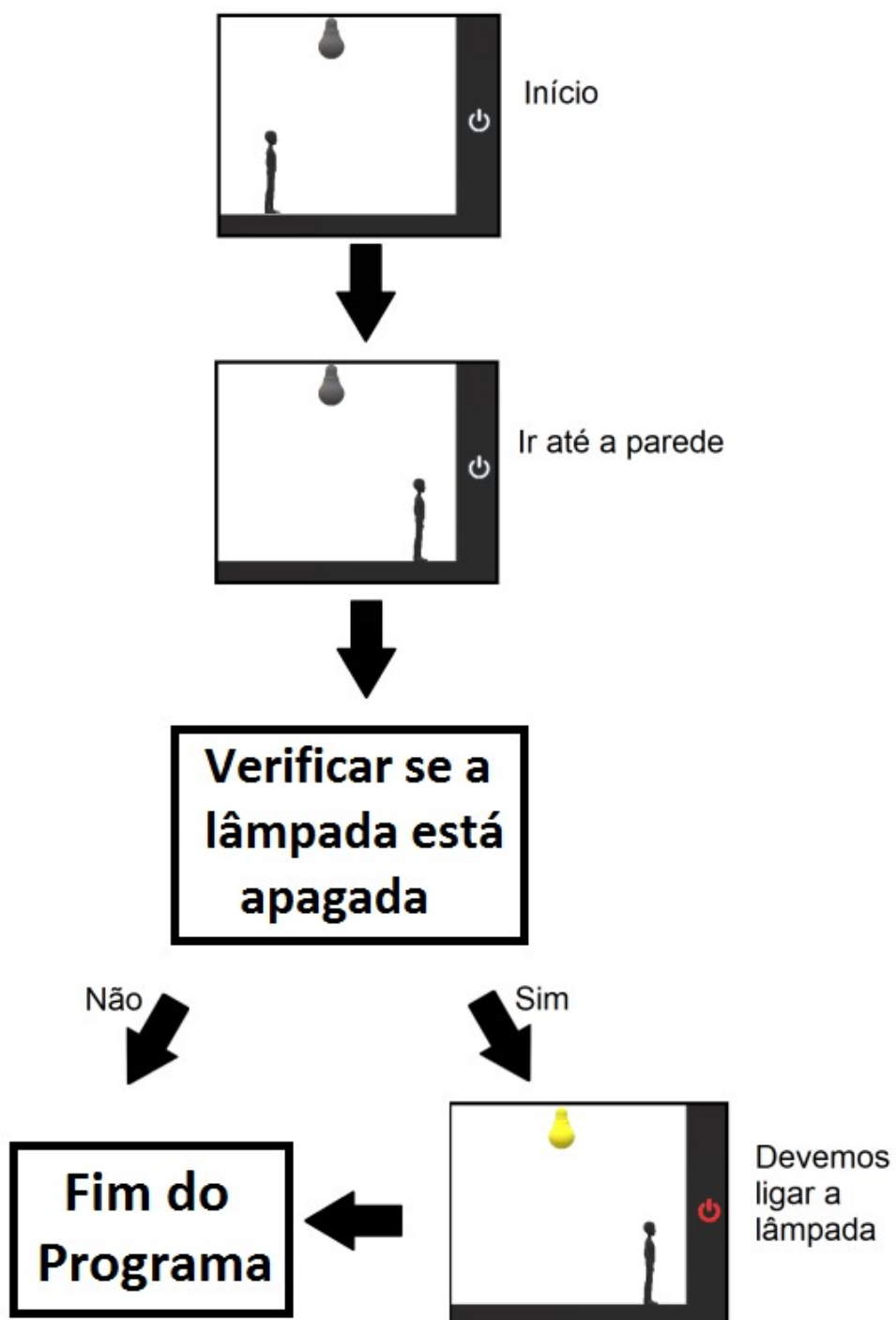


Fonte: Autor (2018).

Um programa possui uma sequência lógica para execução de uma tarefa, o nome que damos para essa sequência é Algoritmo<sup>2</sup>. Nas figuras 2 e 3, são ilustrados exemplos para a realização de uma atividade diária, acender uma lâmpada, com explicitação dos passos necessários para exemplificar a estrutura de um algoritmo.

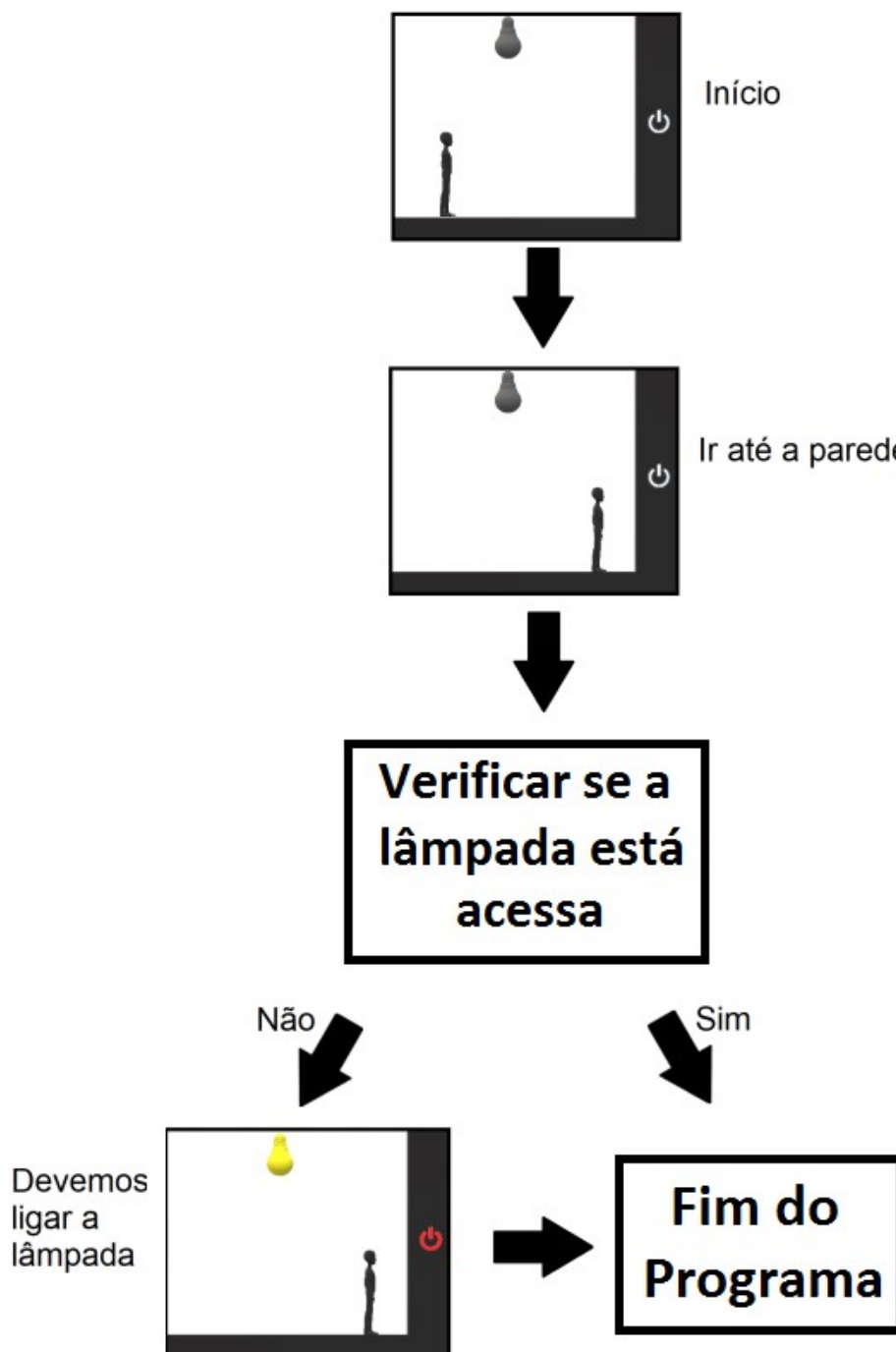
<sup>2</sup>O dicionário Infopédia: Dicionários Porto Editora (disponível em: <https://www.infopedia.pt/dicionarios/lingua-portuguesa/algoritmo>. Acesso em 11 de novembro de 2018) apresenta as seguintes definições para a palavra algoritmo: 1) conjunto de operações predefinidas a seguir de forma sistemática para a resolução de um determinado tipo de problema; 2) MATEMÁTICA conjunto ordenado e finito de processos necessários para efetuar um cálculo; e 3) INFORMÁTICA conjunto de operações, sequenciais, lógicas e não ambíguas, que, aplicadas a um conjunto de dados, permitem encontrar a solução para um problema num número finito de passos.

Figura 2 – Exemplo de passos de um algoritmo para acender uma lâmpada  
(Situação I).



Fonte: Autor (2018).

Figura 3 – Exemplo de passos de um algoritmo para acender uma lâmpada  
(Situação II).



Fonte: Autor.

## 2.2 Estruturas de seleção e operadores lógicos

Em programação precisamos de estruturas de condição (“se” e “senão”) e dos operadores lógicos (“e” e “ou”).



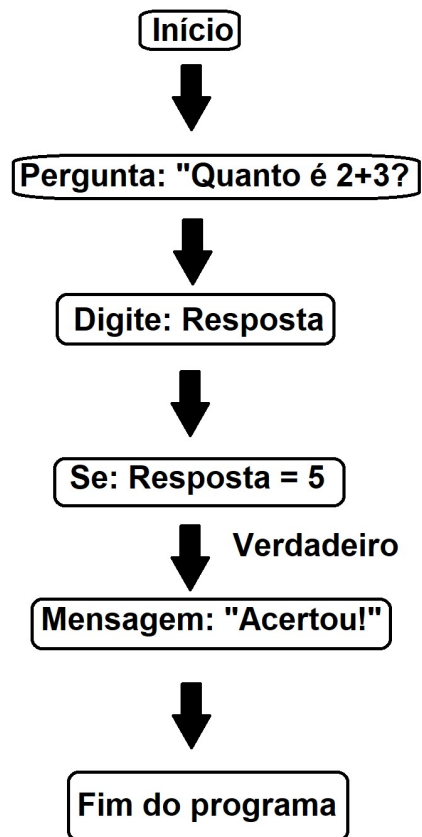
### 2.2.1 Estruturas de Condição “Se” e “Senão”

A estrutura de condição “Se” é utilizada quando precisamos fazer uma comparação de um parâmetro e verificar se este é VERDADEIRO ou FALSO. Caso este parâmetro seja verdadeiro o programa executa um comando específico, caso seja falso, ele simplesmente irá seguir o próximo passo do programa, a menos que exista uma estrutura de condição “*senão*”. Já a estrutura “*Senão*” é utilizada junto com a condição “Se”, sendo que, quando o parâmetro que escolhermos da estrutura “Se” for FALSO, podemos programar para executar um outro comando que escolhermos.

As estruturas de condição podem ser um pouco confusas no começo, mas não se assuste, olhe os exemplos que serão apresentados a seguir.

**Exemplo 1:** Neste exemplo uma pessoa deve responder à pergunta de quanto é  $3+2$  e se a resposta estiver certa o computador dará uma mensagem que a pessoa acertou, observe a Figura 4.

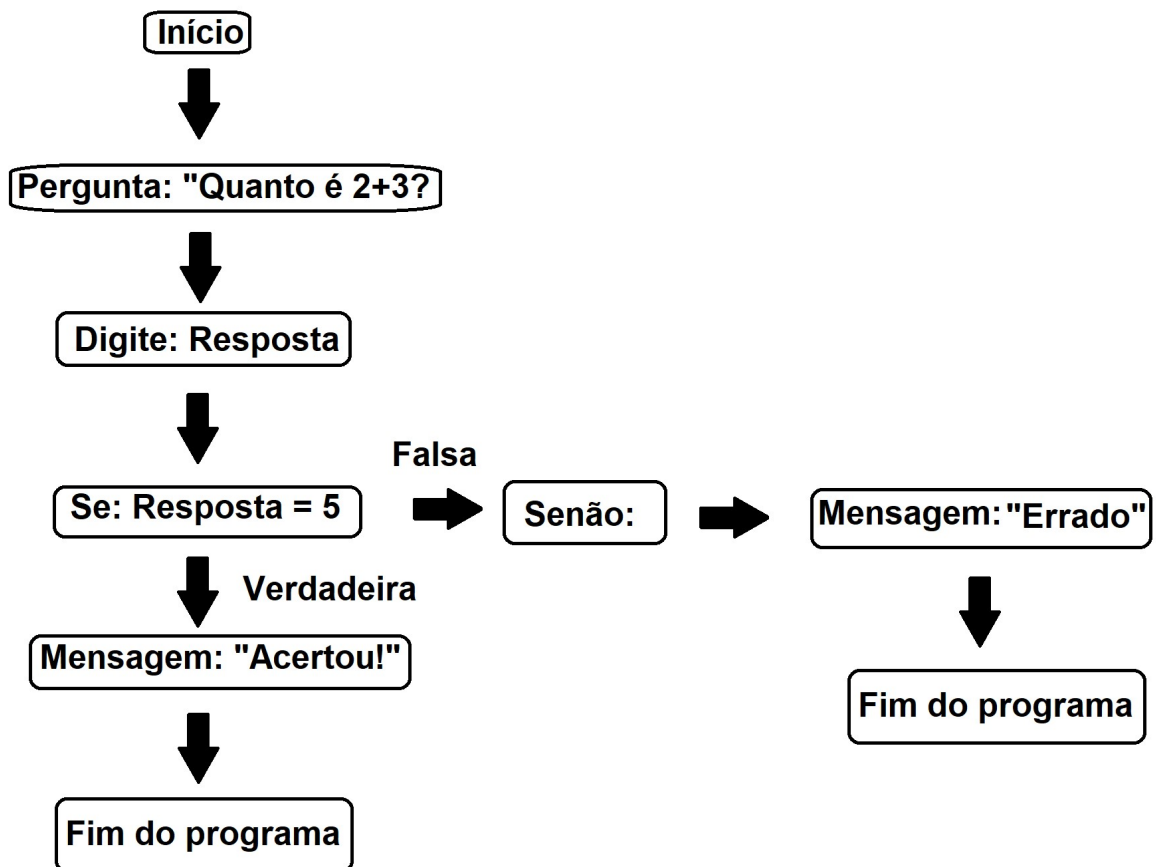
Figura 4 – Exemplo 1: estrutura de condição “Se”.



Fonte: Autor.

**Exemplo 2:** Agora iremos utilizar o mesmo exemplo anterior, mas com uma diferença: o programa irá dar uma mensagem caso a pessoa não acerte a pergunta, conforme pode ser visto na Figura 5.

Figura 5 – Exemplo 2 Estrutura de condição “Se” e “Senão”



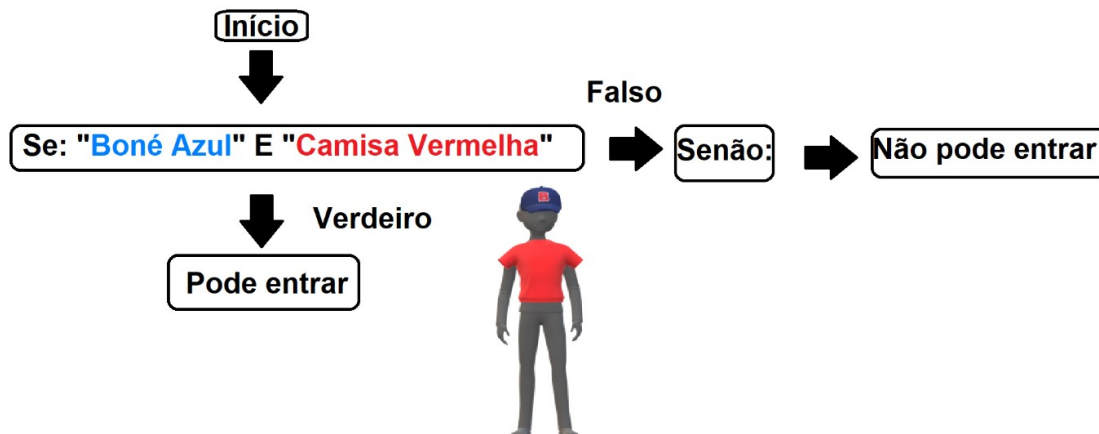
Fonte: Autor.

### 2.2.2 Operadores Lógicos “e” e “ou”

O operador “e” tem função “restritiva”, isto é, existem dois ou mais parâmetros e todos devem ser VERDADEIROS para que o programa execute um determinado passo.

**Exemplo 3:** Você quer entrar em uma festa de aniversário, mas para entrar você precisa estar vestindo um boné azul “e” uma camisa vermelha, ou seja, você precisa estar usando essas duas vestimentas, conforme a Figura 6.

Figura 6 – Exemplo operador “e”.



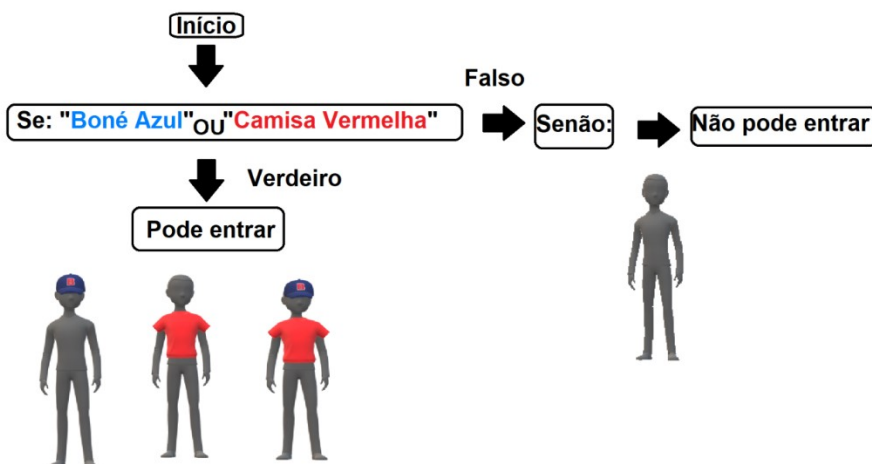
Fonte: Autor.

É necessário estar com o boné azul e a camisa vermelha para poder entrar na festa.

O operador “ou” tem função aditiva, isto é, existem dois ou mais parâmetros e apenas um destes precisa ser VERDADEIRO para que o programa execute um determinado passo.

**Exemplo 4** – Usando o mesmo exemplo anterior, mas agora para você entrar é necessário estar vestindo um boné azul “ou” uma camisa vermelha, ou seja, você precisa estar usando apenas uma das peças de roupa pedida.

Figura 7 – Exemplo operado “ou”



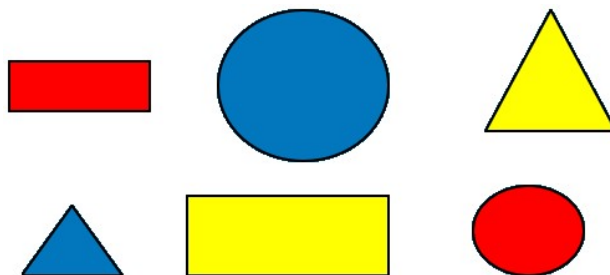
Fonte: Autor (2018).

Como foi possível ver na Figura 12, se você estiver de boné azul ou de camisa de vermelha pode entrar ou com ambos você também pode entrar. Caso você não esteja com nenhuma das duas peças, você não poderá entrar.

### 2.2.3 Blocos Lógicos

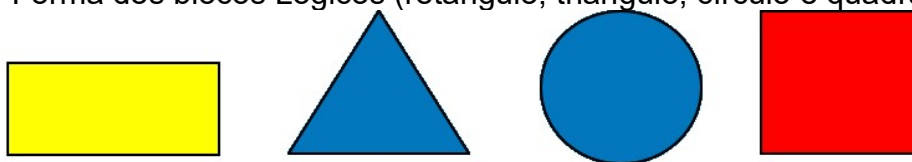
Os blocos lógicos são objetos geométricos (círculos, quadrados, retângulos e triângulos) que nos ajudam a compreender alguns conceitos básicos de programação. Estes objetos possuem características próprias como, por exemplo: forma (Figura 4), cor (Figura 5), tamanho (Figura 6) e espessura (Figura 7).

Figura 8 – Blocos Lógicos.



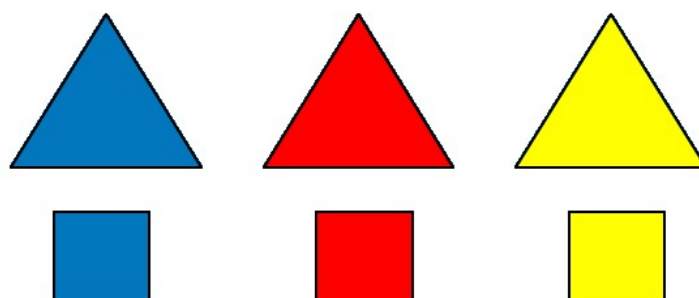
Fonte: Adaptado de PIBID FÍSICA-UNIPAMPA.

Figura 9 – Forma dos blocos Lógicos (retângulo, triângulo, círculo e quadrado).



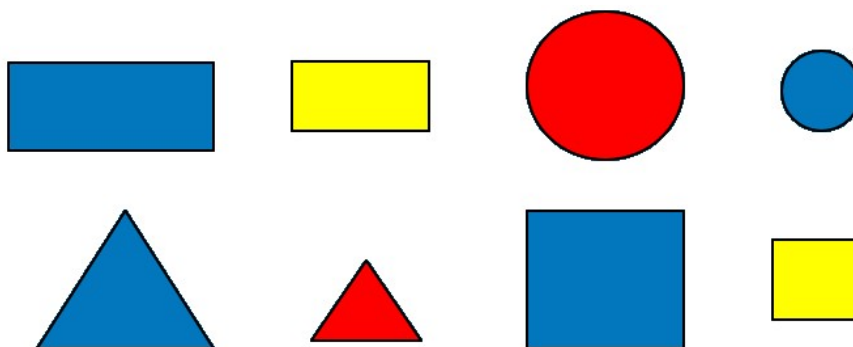
Fonte: Adaptado de PIBID FÍSICA-UNIPAMPA.

Figura 10 – Cores dos blocos de Lógicos (azul, vermelho e amarelo).



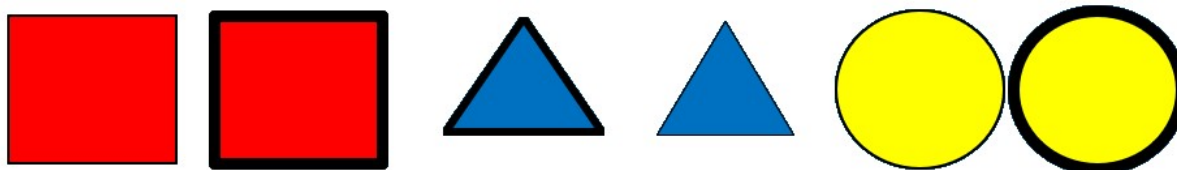
Fonte: Adaptado de PIBID FÍSICA-UNIPAMPA. Disponível em: [https://drive.google.com/file/d/0B\\_7N9cneAZfJTV9NbnZJbUZhQIE/view?usp=sharing](https://drive.google.com/file/d/0B_7N9cneAZfJTV9NbnZJbUZhQIE/view?usp=sharing).

Figura 11 – Tamanhos dos blocos lógicos (grandes e pequenos).



Fonte: Adaptado de PIBID FÍSICA-UNIPAMPA.

Figura 12 – Espessura dos Blocos lógicos (fina e grossa).



Fonte: Adaptado de PIBID FÍSICA-UNIPAMPA.

## 2.2 O que é o Arduino?

De acordo com o site oficial do Arduino:

“O Arduino é uma plataforma de prototipagem eletrônica open-source que se baseia em hardware e software flexíveis e fáceis de usar. É destinado a artistas, designers, hobbistas e qualquer pessoa interessada em criar objetos ou ambientes interativos. O Arduino pode sentir o estado do ambiente que o cerca por meio da recepção de sinais de sensores e pode interagir com os seus arredores, controlando luzes, motores e outros atuadores. O microcontrolador na placa é programado com a linguagem de programação Arduino, baseada na linguagem Wiring, e o ambiente de desenvolvimento Arduino, baseado no ambiente Processing. Os projetos desenvolvidos com o Arduino podem ser autônomos ou podem comunicar-se com um computador para a realização da tarefa, com uso de software específico (ex: Flash, Processing, MaxMSP).”<sup>3</sup>

Além da citação acima, podemos descrever o Arduino como uma plataforma de testes para construção de produtos finais. A placa Arduino é projetada com um microcontrolador da Atmel AVR, com pinos digitais e analógicos para automação de projetos, seu conceito é de *hardware* e *software* livre, ou seja, permite que o usuário faça modificações e construa sua própria placa ou *software* baseado na plataforma do Arduino, o que além de tornar o seu custo mais barato, também

<sup>3</sup> Fonte: <https://playground.arduino.cc/Portugues/HomePage> Acesso: 11/10/2018.

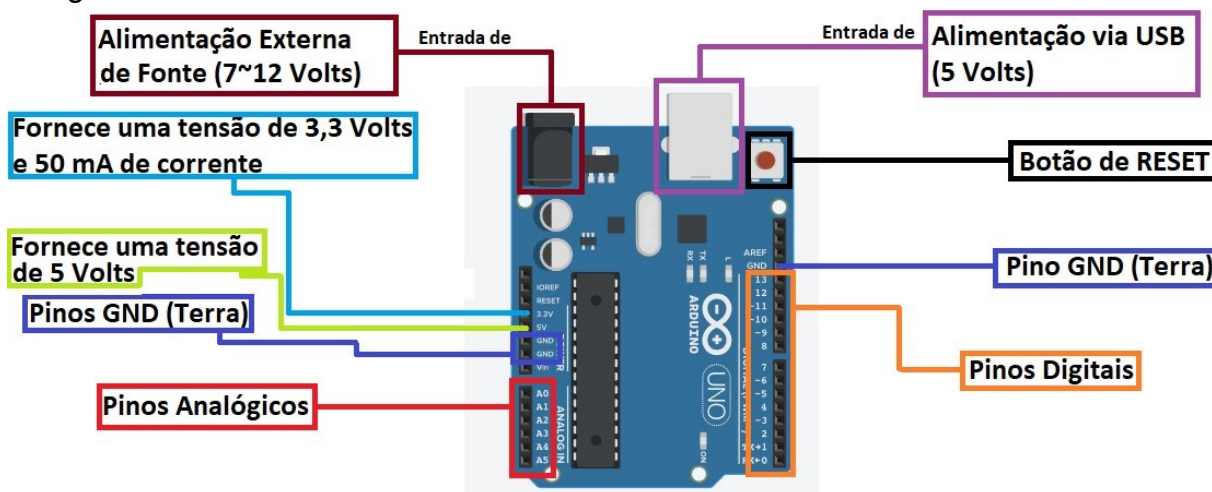
propicia a criação de uma comunidade de usuários bastante ampla que se auxiliam em seus projetos tanto na construção do circuitos quanto na programação.

A placa que iremos utilizar neste minicurso é o Arduino Uno, tem como principais características:

- 6 pinos analógicos;
- 14 pinos digitais;
- 1 Pino de que fornece uma tensão de 5 Volts;
- 1 pino que fornece uma tensão de 3,3 Volts e 50 mA de corrente;
- 3 pinos GND (terra);
- 1 entrada de alimentação via USB de 5 Volts;
- 1 entrada de alimentação externa para baterias 7~12 Volts;
- 1 Botão de *RESET* (Reinicia a programação a programação do Arduino).

Observe a Figura 13 para ver onde ficam as características mencionadas acima:

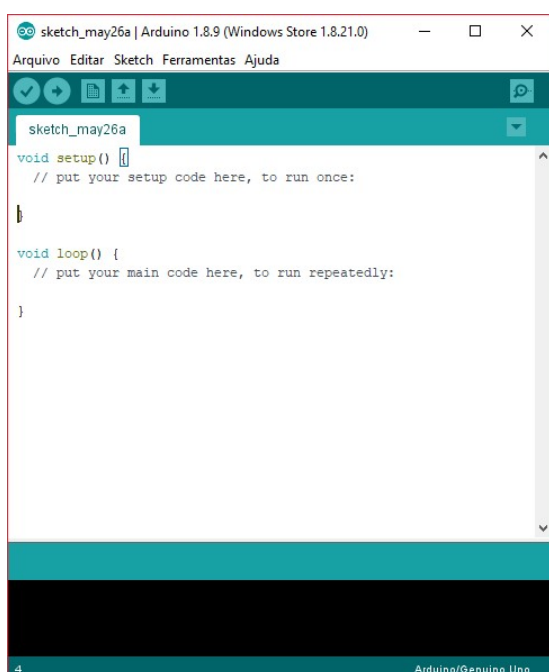
Figura 13 – Placa Arduino Uno



Fonte: Autor

O ambiente de programação de uma placa arduino é no Arduino IDE, que é uma aplicação em Java derivada dos projetos *Processing* e *Wiring*, sua programação é feita através de linhas de comando, entretanto, não iremos nos aprofundar neste estudo, pois o objetivo aqui é trabalhar com a programação em blocos, entretanto, observe a Figura 14 para ver o ambiente de programação do Arduino IDE.

Figura 14 – Ambiente de programação em linhas no Arduino (Arduino IDE)



Fonte: Autor

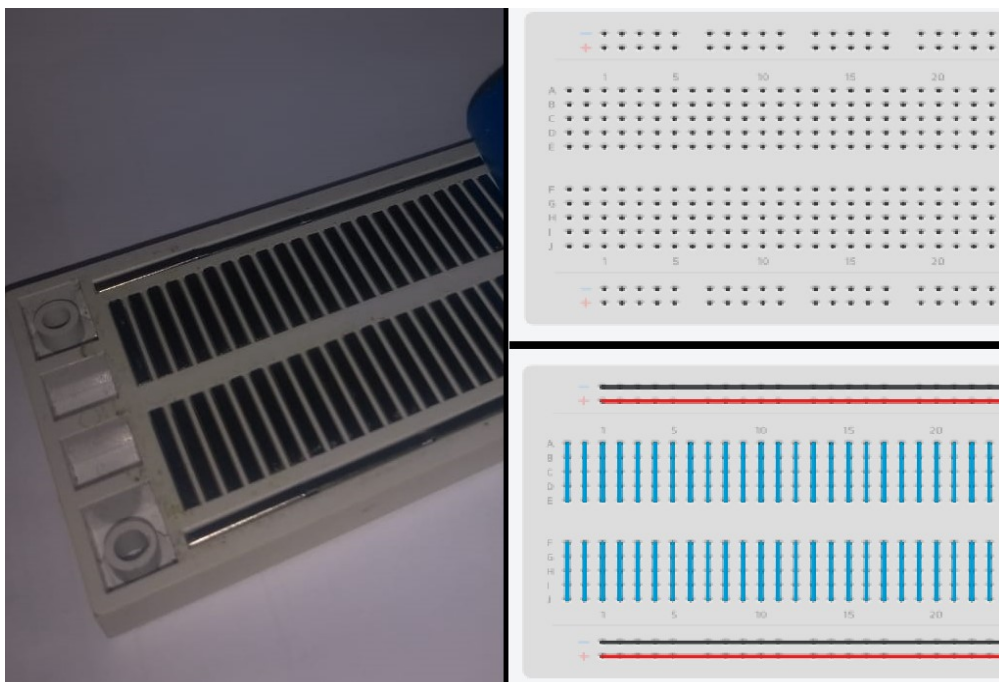
### 2.3 Protoboard

A *protoboard* é uma placa de ensaios experimentais ou também conhecida como placa de prototipagem, ou seja, como o próprio nome diz, é uma placa que permite o teste de circuitos eletrônicos e a criação de protótipos, sem a necessidade de soldar componentes.

A *Protoboard* possui vários “furos” organizados na seguinte maneira: o centro (onde uma placa metálica condutora que liga estes “furos” na vertical) e os extremos (onde tem uma placa metálica que liga estes “furos” na horizontal”). Podemos observar isso melhor na Figura 15.



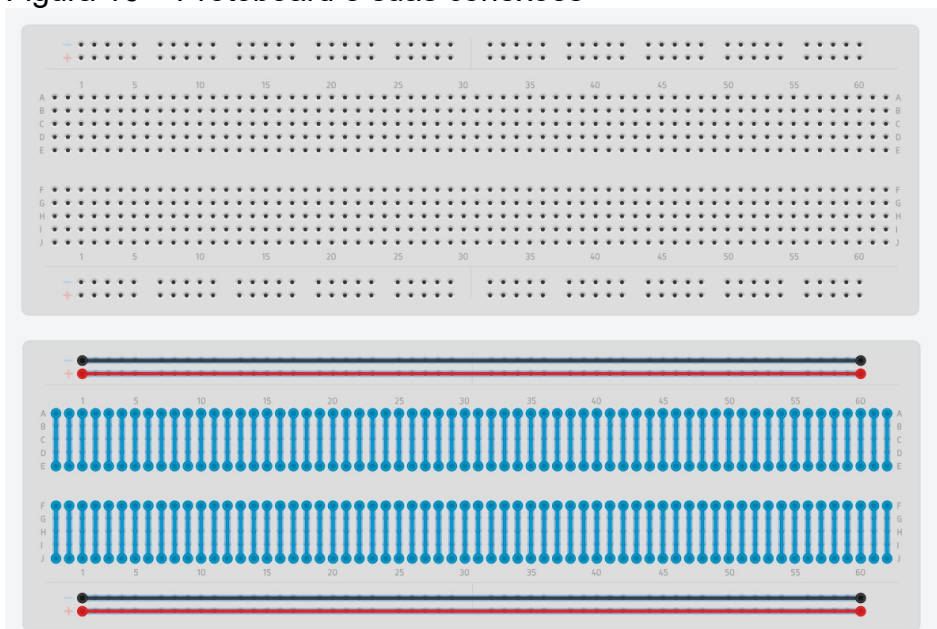
Figura 15 – *Protoboard* e seu funcionamento



Fonte: Autor

A conexão de toda placa *protoboard* pode ser vista na Figura 16.

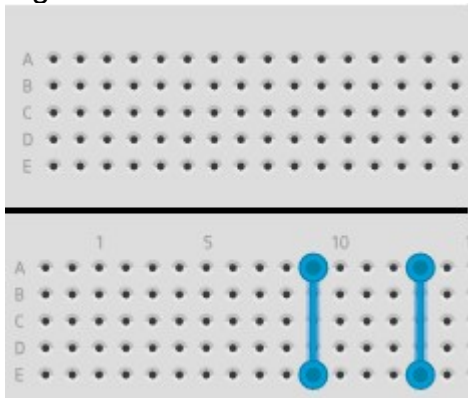
Figura 16 – Protoboard e suas conexões



Fonte: Autor.

Pense agora na seguinte situação: duas dessas linhas de “furos” não estão conectadas entre si, como pode ser vista na Figura 17 (as linhas em azul):

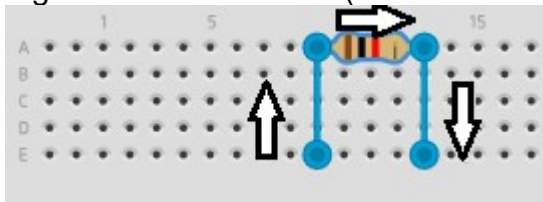
Figura 17 – *Protoboard* conexões duas linhas centrais



Fonte: Autor

Agora, ao conectarmos um resistor entre esses dois terminais, esta passarão a ficar conectados entre si, conforme a Figura 18:

Figura 18 – *Protoboard* (resistor conectando os terminais em azul)



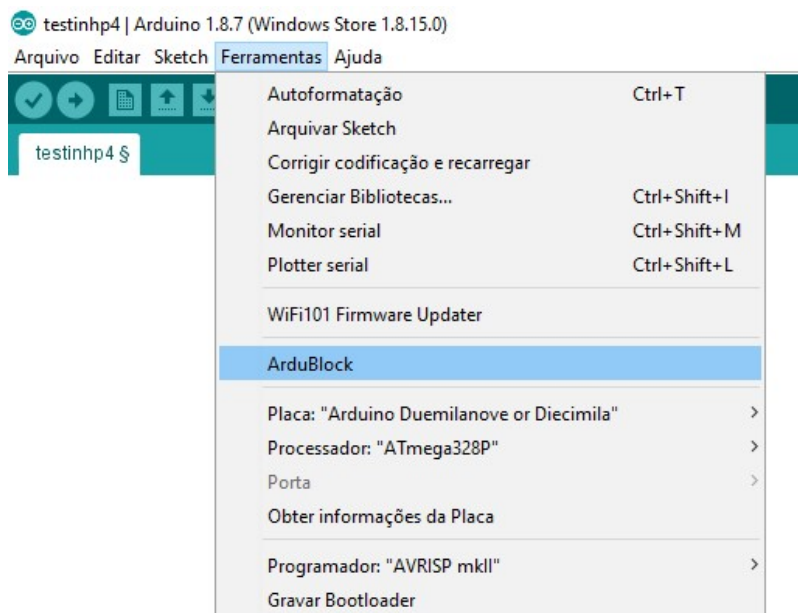
Fonte: Autor.

## 2.4 O que é o Ardublock e qual comandos básicos?

O Ardublock é um aplicativo em Java no qual adicionamos como uma ferramenta no Arduino IDE, sua principal utilidade é no ensino de programação, uma vez que, neste aplicativo, o usuário cria uma programação utilizando blocos e este é transcrito para a linguagem em linhas utilizada na no IDE do Arduino

Para abrir o Ardublock, primeiro abra o Arduino IDE vá em “Ferramentas” e selecione a opção “Ardublock conforme a Figura 19.

Figura 19 – Aba do Ardublock no Arduino IDE.



Fonte: Autor.

A programação no Ardublock, como pode-se observar no próprio nome, é em blocos, ou seja, basta arrastar os blocos (Figura 20) para parte cinza para começarmos a programar.

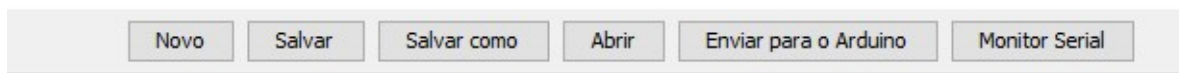
Figura 20 – Arrastar Blocos.



Fonte: Autor.

Como podemos ver na Figura 21, o Ardublock possui alguns comandos básicos.

Figura 21 – Comandos do Ardublock.



Fonte: Autor.

A seguir são descritos brevemente os comandos ilustrados na Figura 21.

**Novo:** Este comando é para quando necessitamos iniciar uma nova programação no Ardublock, ou seja, criar um novo programa.

**Salvar:** Serve para salvar a programação que está sendo realizada no Ardublock.

**Salvar Como:** Se assemelha ao comando anterior com a diferença de que, é possível salvar em outro formato e/ou com outro nome.

**Abrir:** Serve para abrir programações que foram salvas no computador.

**Enviar para o Arduino:** Este é um dos comandos mais importante que tem como função converter a linguagem em blocos para a linguagem em linhas de comando (C++). Após terminar a programação, é necessário enviar o programa para plataforma do Arduino IDE.

**Monitor Serial:** Este comando serve para monitorar a comunicação Serial entre a placa do Arduino e o computador.

## 2.5 Guia de Atividades do Capítulo II

Iremos trabalhar duas atividades referentes ao Capítulo II, a primeira delas é o bingo lógico cujo objetivo é auxiliar no desenvolvimento da lógica computacional. A segunda atividade é um primeiro contato com a montagem de circuitos utilizando a *protoboard* e com a linguagem em blocos do Ardublock, esta atividade não tem

como objetivo o domínio completo da linguagem em blocos, mas sim uma familiarização com a plataforma.

### 2.6.1 Bingo Lógico

O bingo lógico é uma atividade que utiliza os blocos lógicos, tendo como objetivo a abordagem da estrutura de condição “Se” e os operadores lógicos. As cartelas do bingo estão presentes nesta apostila no ANEXO B.

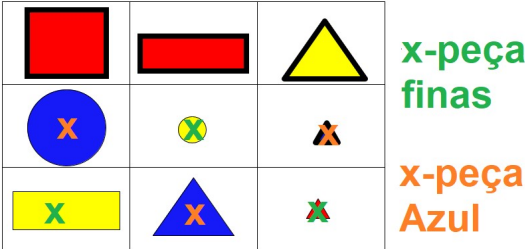
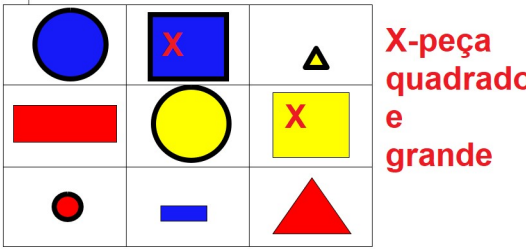
As principais informações sobre regras o bingo são as seguintes:

- 1 - São utilizadas as estruturas de condição “Se”;
- 2 – É feito o sorteio de dois parâmetros. Os sorteios podem ser de: formas, cores, tamanhos ou espessuras;
- 3 – Deve-se realizar o sorteio de um operador lógico e apenas para condição “Se”.
- 5 - O primeiro que preencher a cartela e gritar “Bingo” vence o jogo.

**Obs:** É importante ressaltar que caso se repita algum parâmetro ou os parâmetros sejam conflitantes, o sorteio será feito novamente. Note que existe um conflito a seguir, pois um bloco não pode ser azul e amarelo ao mesmo tempo. No quadro 1, são ilustrados dois exemplos de sorteios.

**Exemplo de conflito:** “Se Parâmetro 1 [Cor azul] e Parâmetro 2 [Cor: Amarela]  
→selecione os blocos [azuis e amarelos]”

Quadro 1 – Exemplos de sorteios.

|  |   |
|--|---|
| Exemplo 1:   | Exemplo 2:  |
| Foi sorteado para o primeiro parâmetro [Cor: Azul] o operador lógico [OU] e o segundo parâmetro foi [Espessura: Fino]. | Foi sorteado para o primeiro parâmetro [Forma: Quadrada] o operador lógico [E] e o segundo parâmetro foi [Tamanho: Grande]. |
| <b>Se:</b> [Bloco Azul OU Fino] fazer:   | <b>Se:</b> [Bloco Quadrado E Grande] fazer:   |
| Selecionar todos blocos azuis <b>OU</b> Finos  | Selecionar todos os blocos quadrados <b>E</b> grandes   |
|                                      |    |

Fonte: Autor (2018).

## 2.6.2 Blink (Acender e apagar um LED)

Um dos programas mais simples que existem na plataforma Arduino é fazer um LED (*Light Emitting Diode* - Tradução: Diodo Emissor de Luz) piscar, sendo que a própria placa Arduino possui um embutido ligado a um pino digital (normalmente ao pino 13) com a finalidade de realizar alguns testes, entretanto, para ficar mais claro como é o funcionamento do *blink* (nome dado a esse código) iremos montar um circuito externo à placa. A programação será feita na extensão Ardublock para plataforma Arduino IDE.

### 2.6.2.1 Montagem do Circuito

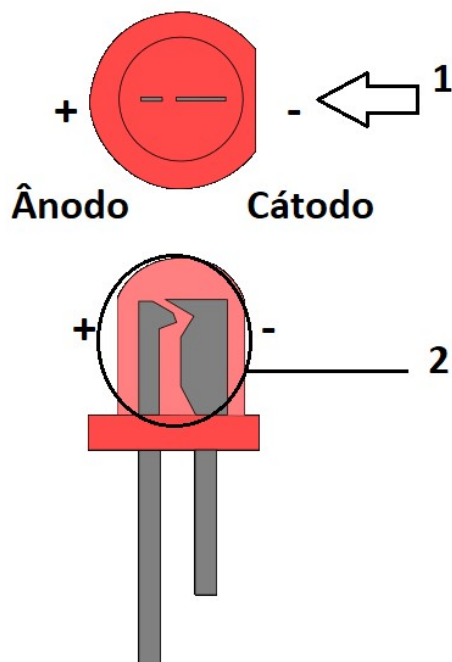
Como já vimos o funcionamento de uma placa *protoboard*, iremos montar nosso circuito nela. O material que iremos utilizar são os seguintes:

- 1 LED;

- 1 Resistor de 100  $\Omega$ ;
- 2 *Jumpers*;
- 1 placa *protoboard*.

Um LED tem dois terminais, um ânodo (+) e um cátodo (-) e existe uma posição correta para conectar estes terminais do LED, observe na Figura 22:

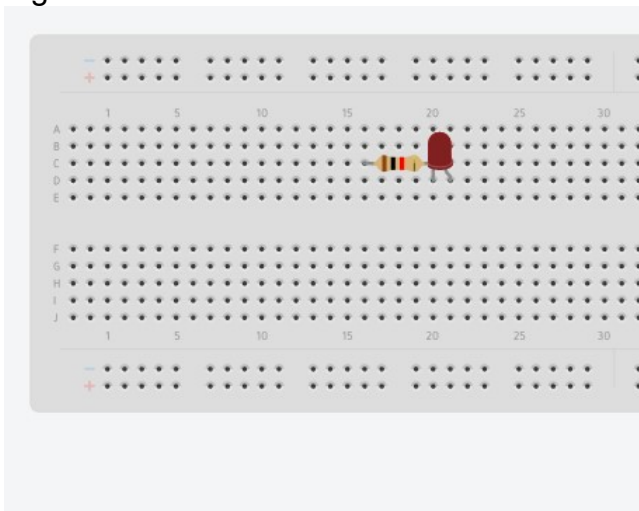
Figura 22 – LED (Ânodo e Cátodo)



Fonte: [encurtador.com.br/EFGMR](http://encurtador.com.br/EFGMR) (Modificado pelo Autor)

Como podemos ver na Figura 22, podemos identificar os terminais de duas formas, sendo elas: 1 – Olhando na parte externa do LED na sua “capsula” qual das partes tem uma marca plana e 2 – Olhando dentro do LED procurando qual é a maior e a menor placa metálica. Na *protoboard* faça a ligação do resistor com o LED conforme a Figura 23.

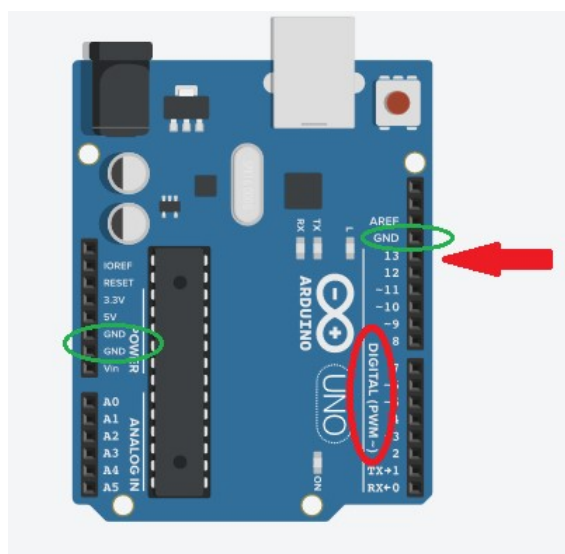
Figura 23 – Conexão LED.



Fonte: Autor.

Iremos utilizar os pinos digitais, estes só podem estar em dois estados ALTO (LIGADO) e BAIXO (DESLIGADO), mas para termos um circuito precisamos também ter o fio terra que no Arduino é o GND. Olhe a Figura 24 para identificar o pino digital 13 e o GND, sendo estes que iremos utilizar. No Arduino temos pinos digitais e pinos analógicos, que serão explicados melhor no decorrer da apostila.

Figura 24 – Identificação dos pinos digital 13 e o GND.

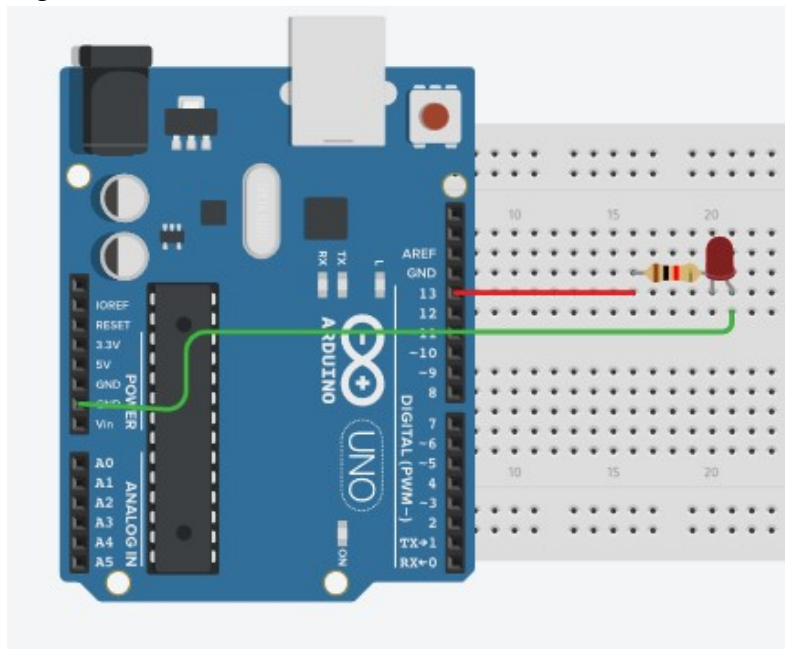


Fonte: Autor



Em verde temos a identificação dos pinos GND e em vermelho a identificação do pino 13 (que fica logo abaixo do GND do lado direito). Conecte um fio no GND e outro no Pino 13 conforme a Figura 25.

Figura 25 – Conexões do circuito.



Fonte: Autor.

### 2.6.2.2 Programação

Vamos pensar no nosso problema, se quisermos fazer uma lâmpada piscar devemos fazer o que? Nada mais é do que acender e apagar uma lâmpada, mas como fazemos isso no Ardublock? Procure interpretar a Figura 26 e a partir dela e dos conhecimentos adquiridos ao longo dessa primeira aula procure montar um projeto que faça um LED ficar 1 segundo aceso e 1 segundo apagado.

Figura 26 – Programa em Ardublock.



Fonte: Autor.

## Capítulo III - Explorando Recursos

### 3.1 Explorando mais comandos do Ardublock

As principais abas do Ardublock podem ser vistas na Figura 27 e são estas que iremos explorar no decorrer desta apostila.

Figura 27 – Abas principais do Ardublock.



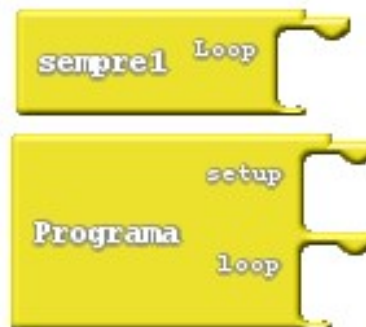
Fonte: Autor.

#### 3.1.1 Aba Controle

A aba controle é a primeira aba e se localiza no canto superior esquerdo, em cor amarela.

##### 3.1.1.1 Comandos para iniciar um programa

Figura 28 – Comandos Iniciais



Fonte: Autor

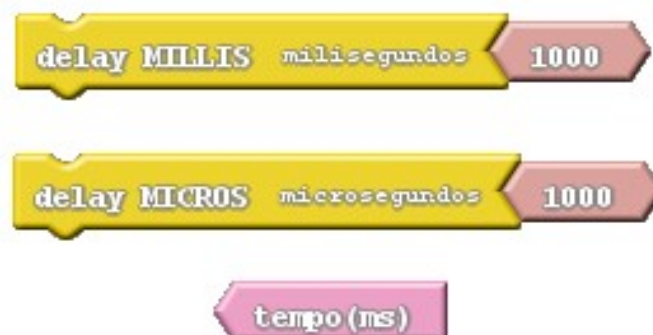
Os comandos *Sempre* e *Programa* são comandos para iniciar uma programação no Ardublock, eles são similares, entretanto, será explicado como cada um funciona.

O comando *Programa* tem duas entradas para blocos, o *Setup* e *Loop*. O *Setup* é executado uma única vez, quando a placa Arduino é iniciada ou resetada e serve para declaração de variáveis, indicação de bibliotecas e estados iniciais dos pinos. O comando *Loop* é utilizado para sempre repetir os mesmos processos, ou seja, garantindo de forma dinâmica e ativa a placa Arduino.

O comando *Sempre* tem uma única entrada que é *Loop* e tem a mesma função que no comando *Programa*<sup>4</sup>, que é utilizado para sempre repetir os mesmos processos.

### 3.1.1.2 Comandos de tempo

Figura 29 – Comandos de tempo



Fonte: Autor

O comando *delay MILLIS* tem como função esperar um certo tempo antes de realizar o próximo passo de um programa, conforme a Figura 29 a parte em rosa é igual a 1000 (milissegundos), e este valor pode ser alterado. O valor de 1000 milissegundos = 1 segundo. Exemplo: no código de *blink* (fazer um LED piscar) que foi visto anteriormente, ligamos o pino digital, usamos o código de *delay*, desligamos o pino digital e vamos e utilizar novamente um comando *delay*.

Tabela 1 – Conversão de tempo

|           |                       |
|-----------|-----------------------|
| 1 segundo | 1000 milissegundos    |
| 1 minuto  | 60.000 milissegundos  |
| 1 hora    | 360.000 milissegundos |

Assim como o *delay MILLIS*, o comando *delay MICROS* também tem como função esperar um certo tempo antes de realizar o próximo passo de um programa, entretanto a diferença está na unidade, o *delay MICROS* está na unidade de microssegundos, ou seja, 1.000.000 microssegundos = 1 segundo. O bloco *tempo(ms)* é uma variável inteira que conta o tempo em uma unidade de milissegundos (1000 milissegundos = 1 segundo). Com esse bloco, por exemplo,

<sup>4</sup> Uma observação importante em relação a esses dois comandos é que, apesar de no comando *Programa* termos a opção de declarar as variáveis inicialmente e explicitar as bibliotecas, isto não é necessário, uma vez que, o próprio Ardublock faz a declaração de variáveis e as bibliotecas quando transcreve da linguagem em blocos para linguagem do Arduino IDE, assim sendo, a principal utilidade do comando *Setup* é de estabelecer os valores iniciais dos pinos do Arduino pinos. Podemos concluir também que o comando *Sempre* pode ser utilizado sempre que não for necessário estabelecer os valores iniciais dos pinos.

conseguimos medir a velocidade de um objeto, pois a velocidade é dependente do tempo, ou seja, nós precisamos medir<sup>5</sup> o tempo para descobrir a velocidade de um objeto.

### 3.1.1.3 Estruturas de seleção e Aba de Testes

Iremos trabalhar as estruturas de seleção no Ardublock, entretanto, precisamos trabalhar também aba de testes, pois estas estão extremamente interligadas, uma vez que, para explorarmos uma estrutura de seleção precisamos de um parâmetro de comparação. Iremos trabalhar apenas com duas estruturas de condição, se e senão.

Figura 30 – Estrutura de Condição



Fonte: Autor

Já vimos, no Capítulo II, as estruturas de seleção, entretanto, iremos aborda-las utilizando o Ardublock.







Podemos dizer que o comando “Se” funciona como um bloco de testes, colocamos uma condição no Ardublock conhecida como teste, e caso esta seja satisfeita (seja verdadeira), o programa irá realizar um determinado passo.

O comando “Se/Senão” assemelha-se com a condição “Se”, entretanto caso o teste da condição “Se” não seja satisfeito, o programa irá realizar um outro passo especificado pelo programador.

As principais condições da aba “teste” são apresentadas no Quadro 2:

<sup>5</sup> A velocidade referida neste trecho é a velocidade média, ou seja,  $V = \Delta x / \Delta t$ . A velocidade também tem uma dependência em relação a posição, assim sendo, também precisamos medir o deslocamento de um dado objeto para descobrir sua velocidade.



### Quadro 2 – Comparadores

|   |  |
|---|--|
|  | Verdade se o primeiro parâmetro for maior que o segundo.       |
|  | Verdade se o primeiro parâmetro for menor que o segundo.       |
|  | Verdade se o primeiro parâmetro for igual ao segundo.          |
|  | Verdade se o primeiro parâmetro for maior ou igual ao segundo. |
|  | Verdade se o primeiro parâmetro for menor ou igual ao segundo. |
|  | Verdade se o primeiro parâmetro for diferente do segundo.      |

Fonte: Autor

Os operadores que iremos trabalhar são:

### Quadro 3 – Operadores

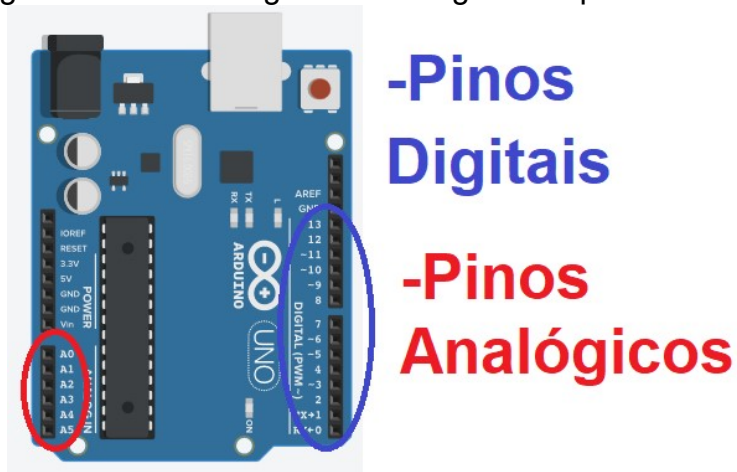
|  |  |
|--|--|
|   | Função restritiva, isto é, é verdade se dois ou mais parâmetros são verdades.              |
|  | Função aditiva, isso é, é verdade quando apenas um dos parâmetros precisa ser uma verdade. |

Fonte: Autor

### 3.1.2 Aba Pinos

No Arduino existem dois tipos de pinos: os analógicos e os digitais. Os pinos digitais têm dois estados possíveis “ALTO” e “BAIXO”, podendo fazer uma analogia como um estado ALTO=Ligado e um estado BAIXO=Desligado. Os pinos analógicos servem para medidas e os valores variam de 0 a 1023. Os pinos digitais e analógicos são identificados na Figura 31:

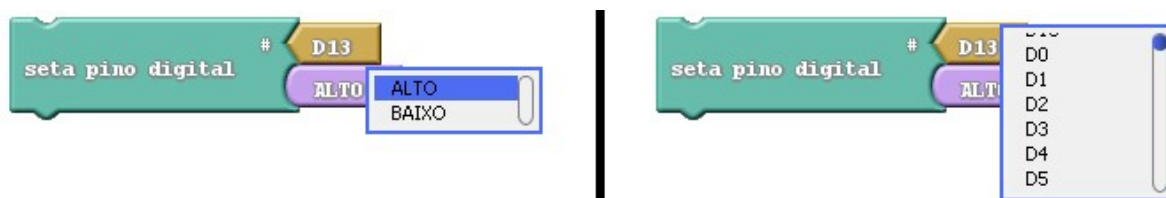
Figura 31 – Pinos Digitais e analógicos na placa Arduino



Fonte: Autor

No bloco que vemos na Figura 32, podemos definir para um pino digital os estados “ALTO” (Ligado) ou “BAIXO” (desligado), e também podemos definir qual dos pinos digitais iremos utilizar, por exemplo, o D13 é o que corresponde ao pino digital 13.

Figura 32 – Pinos e estados dos pinos digitais



Fonte: Autor

Os pinos analógicos servem para medir a diferença de potencial e estes valores variam de 0 a 1023, assim sendo, este irá retornar um inteiro. No ArduBlock existe um bloco que expressa o valor desse pino:

Figura 33-Pino Analógico A0



Fonte: Autor

### 3.1.4 Aba Matemática

Em programação precisamos muitas vezes realizar operações matemáticas. O ArduBlock traz diversas funções, entretanto, as principais que iremos utilizar são:

Quadro 4 – Operações Matemáticas

|  |                |
|--|----------------|
|  | Soma.          |
|  | Subtração.     |
|  | Multiplicação. |
|  | Divisão.       |

Fonte: Autor

### 3.1.5 Aba Variáveis

Figura 34.a – Variáveis inteiras



Fonte: Autor

Figura 34.b – Variáveis decimais



Fonte: Autor

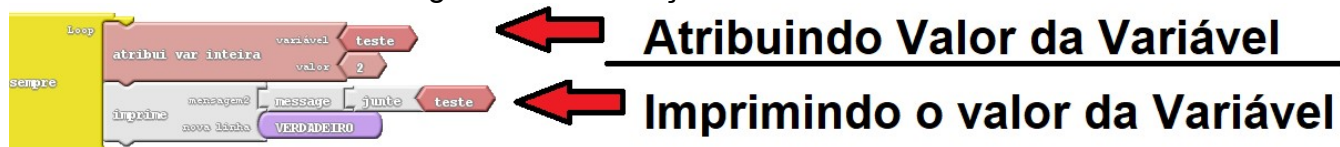
Iremos trabalhar com dois tipos de variáveis, as inteiras e as decimais, mas qual a diferença entre estas duas? Vamos olhar a definição a seguir:

**Variáveis Inteiras:** São números inteiros, ou seja, são números que não têm casas decimais. Exemplo: 125

**Variáveis Decimais:** São números com casas decimas, ou seja, números com vírgulas. Exemplo: 3,14

Podemos atribuir valores para estes dois tipos de variáveis com os comandos que aparecem nas figuras 34.a e 34.b, entretanto, não necessariamente precisamos atribuir um valor, mas sim, podemos escrever uma função. Sempre que vamos “chamar” uma variável teremos que fazer isso pelo nome que dermos a ela, como poderá ser observado na Figura 35, cujo o nome que foi dado a variável foi “teste”.

Figura 35 – Atribuição da variável



Fonte: Autor

### 3.1.7 Aba Comunicação

Sempre que quisermos imprimir uma mensagem, ou seja, ter um valor de retorno utilizamos o comando de “Imprimir” e trocamos para ter um resultado em uma linha após trocamos para verdadeiro o comando abaixo:

Figura 36- Comando imprimir (nova linha)



Fonte: Autor

Se desejarmos, poderemos escrever uma mensagem que aparecerá junto com o nosso valor, modificando de “message” para a mensagem desejada, conforme pode ser visto na Figura 37.



Figura 37- Comando Imprimir (mudando a mensagem)



Fonte: Autor

A última coisa que precisamos fazer para imprimir, por exemplo, uma determinada temperatura, é usar o comando “junte” para juntar o valor da variável “*Temperatura*” com a nossa mensagem, de acordo com a Figura 38.

Figura 38- Comando imprimir (junte)



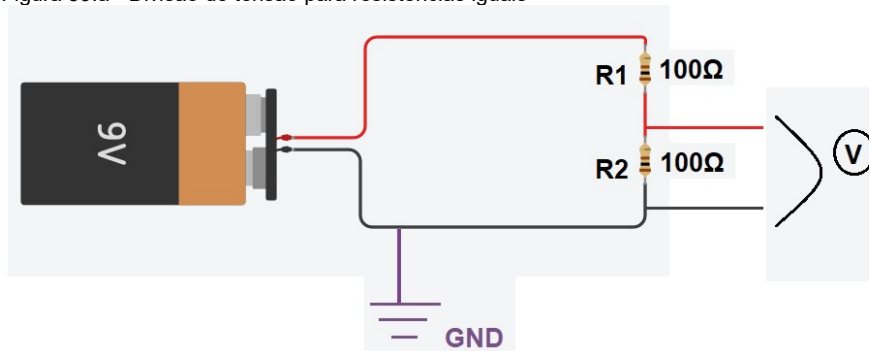
Fonte: Autor

### 3.2 Divisor de tensão

O divisor de tensão é um circuito que faz a divisão da tensão aplicada nas suas extremidades, sendo que, só temos divisão de tensão quando temos a associação de duas resistências. Na Figura 39.a, temos uma representação da divisão de tensão para duas resistências iguais e na Figura 39.b a representação para duas resistências diferentes.



Figura 39.a - Divisão de tensão para resistências iguais

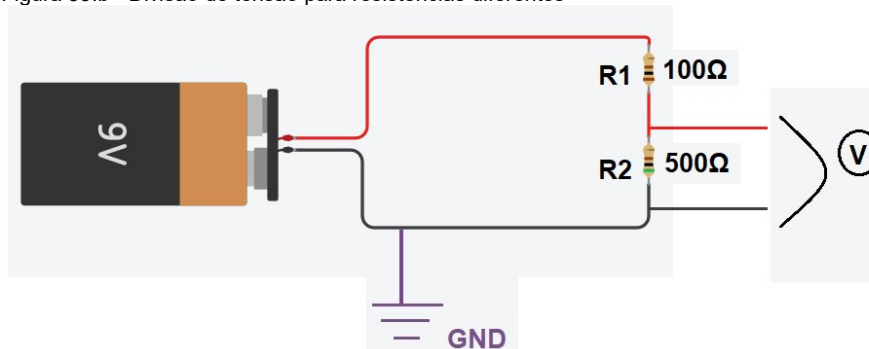
se:  $(R_1 = R_2)$ 

$$V_{out} = \frac{1}{2} V_{in}$$

(Tensão de saída para Resistências iguais)  
Eq. 1

Fonte: Autor

Figura 39.b - Divisão de tensão para resistências diferentes

se:  $(R_1 \neq R_2)$ 

$$V_{out} = \frac{R_2}{(R_1 + R_2)} V_{in}$$

(Tensão de saída para Resistências diferentes)  
Eq. 2

Fonte: Autor

Para entendermos melhor as equações acima, segue a seguinte explicação:  $V_{in}$  é a tensão de entrada, ou seja, no exemplo acima esta tensão será de 9 Volts.  $V_{out}$  é a tensão de saída, ou seja, o valor da queda de tensão devido a presença de resistores. Esta representação está marcada no nosso circuito com um círculo com um **V** no meio.  $R_1$  e  $R_2$  são nossas resistências. Analisando os dois casos, temos:

- Se  $(R_1 = R_2)$ , isso significa que a queda de tensão irá se dividir por igual entre os dois resistores, conforme pode ser observado na Eq. 1.
- Se  $(R_1 \neq R_2)$  caso as resistências sejam diferentes, a tensão não se dividirá por igual, logo, a Eq. 2 descreve como é seu funcionamento.

Agora que vimos o que é a divisão de tensão e como é seu funcionamento, fica a seguinte pergunta: como utilizamos o Arduino para sua medida? E principalmente, qual a sua importância para este minicurso?

Na secção 2.1.3, vimos que um pino digital tem um estado ALTO (ligado) e um estado BAIXO (desligado), mas em termos de tensão, qual seu significado?

Tabela 6- Entradas e estados

| Entradas Digitais |        |         |
|-------------------|--------|---------|
| Estado:           | Alto=1 | Baixo=0 |
| Tensão (V)        | 5      | 0       |

Fonte: Adaptado de Ribeiro (2017)

O Arduino tem uma leitura de 0 a 1023 em suas entradas analógicas, onde 1023 corresponde a uma tensão igual a 5 Volts e o 0 corresponde a uma tensão igual a 0 Volts. Utilizamos as entradas analógicas para fazer a leitura de tensões variadas, que podem ser valores de, no mínimo, 0,0049V entre uma tensão e outra (no caso de trabalharmos com 5V, as subdivisões possíveis são valores de, no mínimo,  $5V \div 1023 = 0,0049V$ ) conforme pode ser visto na tabela abaixo:

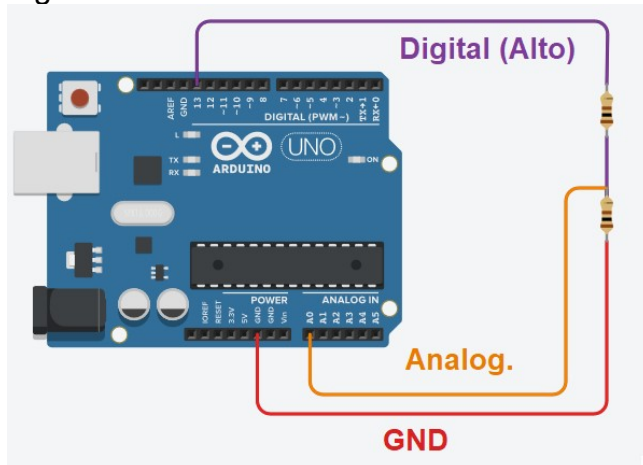
Tabela 7 – Entradas analógicas e a tensão por bits

| Entradas Analógicas |   |        |        |        |        |        |        |     |      |
|---------------------|---|--------|--------|--------|--------|--------|--------|-----|------|
| Tensão(V)           | 0 | 0,0049 | 0,0098 | 0,0147 | 0,0196 | 0,0245 | 0,0294 | ... | 5    |
| Arduino (10 bits)   | 0 | 1      | 2      | 3      | 4      | 5      | 6      | ... | 1023 |

Fonte: Ribeiro (2017)

Isso significa que para 1 de medida (de 0 a 1023) teremos uma variação de valor de 0,0049 Volts.

Figura 40 – Divisor de tensão utilizando Arduino



Fonte: Autor.

Na Figura 40, podemos observar uma alimentação de 5 volts e o fio terra (GND). A entrada analógica mostrará um número inteiro (de 0 a 1023) que representa a

queda de tensão causada pelos dois resistores, deste modo, para calcularmos a queda de tensão temos que realizar uma regra de três.

$$\begin{aligned} 5\text{volts} &\leftrightarrow 1023 \\ V_{\text{out}} &\leftrightarrow A0 \\ V_{\text{out}} &= \frac{5 \cdot A0}{1023} \end{aligned}$$

O Arduino tem a leitura de 0 a 1023, sabendo que quando o arduino tem uma leitura na porta analógica igual 1023 a tensão correspondente é igual a 5 Volts e quando temos uma leitura de 0 temos uma tensão equivalente a 0 Volts, logo  $V_{\text{out}}$  é igual ao valor que está sendo medido na porta analógica (neste caso, a porta analógica 0) multiplicado pela tensão máxima (5 Volts) divididos pela leitura máxima do Arduino (1023).

Mas qual a importância da divisão de tensão para nossos estudos? Bom, temos diversas medidas de temperatura, luminosidade e etc., e a divisão de tensão serve para conseguirmos mensurar a diferença de potencial (em volts) de alguns tipos de sensores que apresentam as mesmas características de resistores variáveis. Como o Arduino não mede resistência elétrica, apenas diferença de potencial (ddp), o divisor de tensão transforma essa variação de resistência em variação de ddp. Utilizaremos sensores para fazer essas medidas, que no caso, é nosso próximo tópico.

### 3.3 Sensores

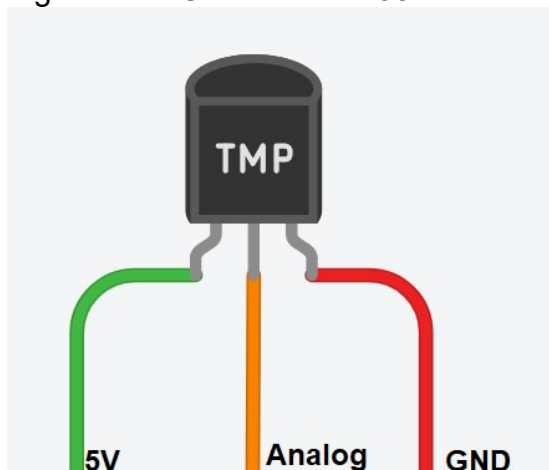
Sensores são dispositivos que servem para obter certos tipos de medidas externas, por exemplo temperatura, pressão, luminosidade, distância, presença de um campo magnético, umidade e etc. Sensores são capazes de modificar suas características automaticamente e podem ser classificados em duas categorias os “*sensores discretos*” e os “*sensores contínuos*”.

- **Sensores discretos** são aqueles que apresentam um único estado, ou seja, ligado e desligado Ex: *reed sweet*.
- **Sensores contínuos** são sensores que monitoram grandezas físicas numa faixa de valores continua e a saída é proporcional a essa grandeza. Ex: LM35 e o LDR.

### 3.3.1 LM 35

O LM35 é um circuito interno cuja finalidade é a medição de temperatura. Para fazer as conexões corretas, veja a para Figura 41. O LM35 deve estar com a face plana virado para você. A alimentação de 5 Volts (Verde) deve estar ligada na extremidade a esquerda, o GND (Vermelho) deve ser ligado no terminal da direita e o terminal do meio é onde fica a conexão com uma porta analógica.

Figura 41 – Conexões LM35



Fonte: Autor.

Como vimos anteriormente no divisor de tensão, temos uma tensão de entrada e uma queda de tensão em uma resistência. Quando o LM35 tem uma variação equivalente a 0,01 Volts temos uma variação de 1°C.

$$\begin{aligned} 5\text{volts} &\leftrightarrow 1023 \\ 0,01 \times T &\leftrightarrow A0 \end{aligned}$$

Sabemos que a tensão de entrada é 5 Volts e seu valor corresponde a 1023. O A0 é o valor que a porta analógica nos informará sobre quanto é o valor que está sendo medido que varia entre 0 e 1023. Este valor será equivalente a tensão de saída do LM35. Sabendo que  $0,01 \times T$  é o equivalente a 0,01 Volts/°C multiplicado pela temperatura. Iremos isolar nossa variável  $T$  pois desejamos encontrar a temperatura, logo:

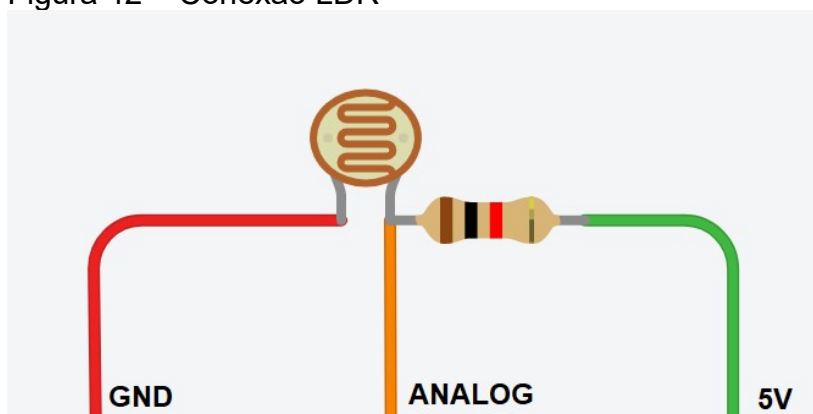
$$1023 \times 0,01 \times T = 5 \times A0$$

$$T = \frac{5 \times A0}{10.23}$$

### 3.3.2 LDR

Um LDR é uma resistência que varia de acordo com a intensidade luminosa, ou seja, sua aplicação é como um sensor para variação de intensidade luminosa. O LDR deve estar ligado em série com a resistência de 3,3k $\Omega$ . Conforme a Figura 42, a extremidade do LDR (Vermelho) deve estar ligada no GND no Arduino, na conexão entre a resistência e o LDR (Laranja) deve estar ligada em uma entrada analógica no Arduino e na extremidade do resistor (Verde) deve estar ligada na entrada de 5 Volts.

Figura 42 – Conexão LDR

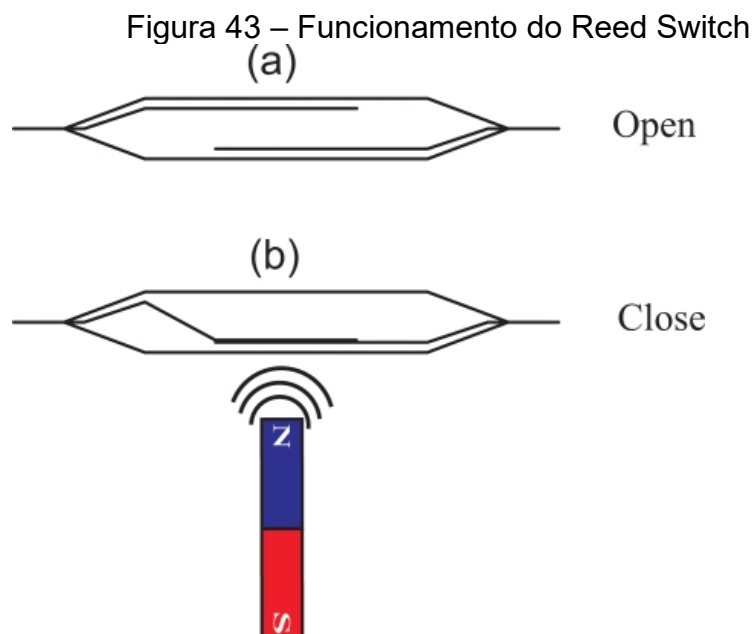


Fonte: Autor.

O valor que nos interessa é a tensão de saída do que vai para o pino analógico, essa variação de tensão vai nos mostrar como é que o valor da resistência do LDR está variando de acordo com a luminosidade.

### 3.3.3 Reed Switch

Um *reed switch* é um dispositivo com duas lâminas flexíveis em um bulbo de vidro e seu funcionamento é o equivalente a um interruptor magnético, ou seja, na presença de um campo magnético o circuito fecha-se permitindo a passagem de corrente como ser visto em (b), caso não exista um campo magnético o circuito fica aberto, ou seja, não permite a passagem de corrente.



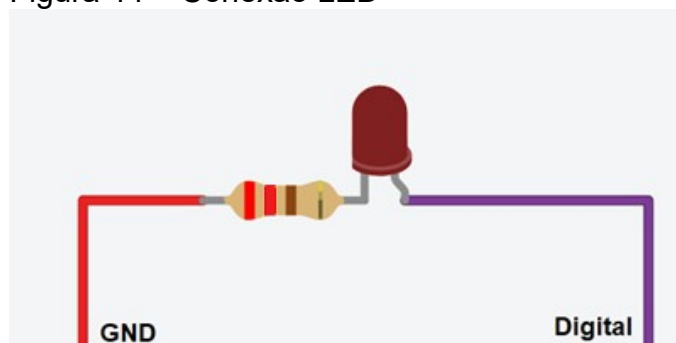
### 3.4 Atuadores

Os atuadores são dispositivos que agem a partir do que foi programado. De acordo com Ribeiro, 2017 “Os atuadores são qualquer tipo de elemento pelo qual o Arduino irá responder externamente, ou seja, podemos dizer que um atuador é a resposta final a partir de parâmetros preestabelecidos por nós através da programação e usando as variáveis verificadas pelos sensores.” Ex: relê, LED, motor e etc.

#### 3.4.1 LED

O LED precisa estar associado em série com uma resistência 220Ω. Conforme a Figura 44 a extremidade com a resistência deve ser ligada no GND e a extremidade “torta” do LED deve ser ligada numa entrada digital.

Figura 44 – Conexão LED

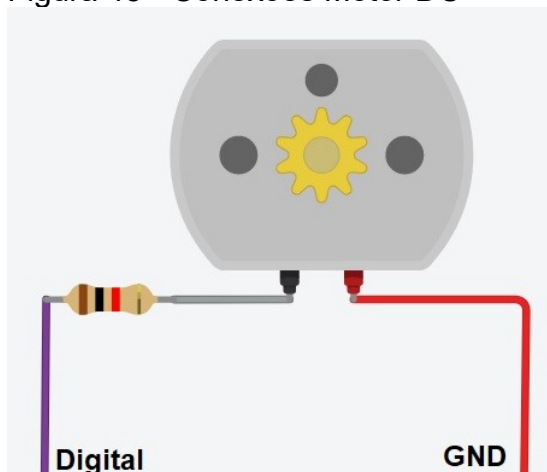


Fonte: Produzido pelo autor utilizando Tinkecard.

### 3.4.2 Motor

Um Motor DC (*Direct Current*) é um dispositivo que transforma energia elétrica em movimento. Deve estar ligado a uma resistência de 100Ω. Na extremidade com a resistência conecta-se com a porta digital (Roxo) e a outra extremidade do Motor conecta-se a porta GND (vermelho).

Figura 45 - Conexões Motor DC

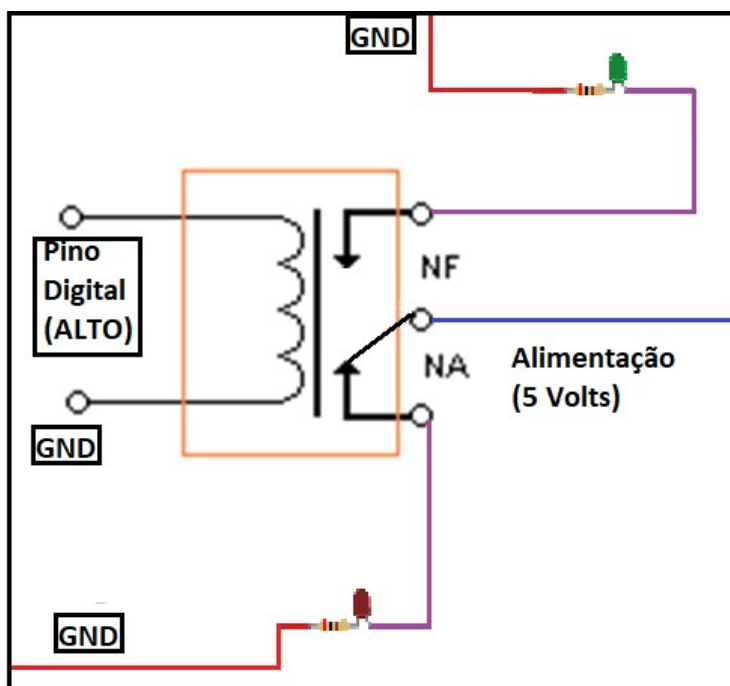


Fonte: Autor

### 3.4.3 Relê

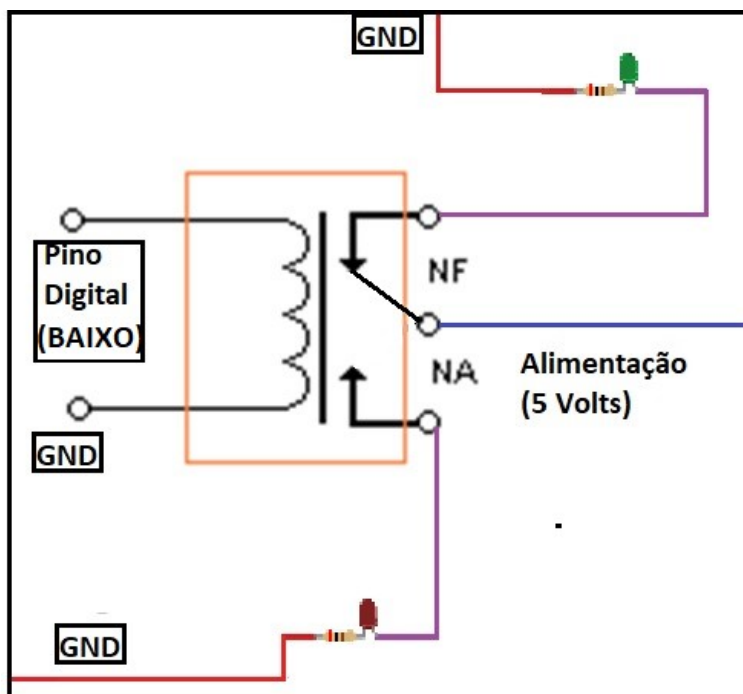
O relê é um dispositivo que funciona como uma chave, entretanto, ao contrário do *reed switch* ele é um atuador, tendo uma bobina que pode ligar uma chave ou outra. A relê abaixo tem um total de 5 terminais (indicado pelas bolinhas) temos dois terminais para ligarmos ou desligarmos a bobina, um no pino digital e outro no pino GND. Existe um terminal que fica na chave e os outros dois terminais restantes ficam ligados nos estados NA e NF (que serão explicados a seguir). Ao ligar o estado alto a chave NA (Normalmente aberto) se fecha. Caso o pino digital esteja no estado baixo, a chave da relê permanece na posição NF (Normalmente Fechado). Imagine que você tem dois LED's, um verde e um vermelho, e quer deixar um deles ligado em quanto o outro fica desligado, temos os dois pinos para ligar ou desligar a bobina. Ao definirmos um estado ALTO (Figura 46.a) deixamos a bobina ligada e a posição que era NA fica fechada, fazendo com que, LED vermelho acenda. Se definirmos o estado um estado BAIXO (Figura 46.b), a bobina se desliga e a chave vai para o seu estado normal (NF), fazendo com que, o LED verde acenda.

Figura 46.a – Relê (Estado Alto)



Fonte: Retirado (<https://athoselectronics.com/wp-content/uploads/2016/01/SimbologiaRele.gif>) em 26/05/2019 e modificado pelo autor desta apostila

Figura 46.b – Relê (Estado Baixo)



Fonte: Retirado (<https://athoselectronics.com/wp-content/uploads/2016/01/SimbologiaRele.gif>) em 26/05/2019 e modificado pelo autor desta apostila



### 3.5 Atividades do Capítulo III

#### 3.5.1 Construindo um Cronômetro

Vamos construir um cronômetro, cujo objetivo é medir o número de segundos e milésimos de segundos. Este programa pode ser muito utilizado na medida do tempo para, por exemplo, determinar a velocidade média ( $v_{med} = \Delta S / \Delta t$ ) que objeto tem. O único material utilizado será uma placa Arduino.

Dicas:

- Um cronômetro começa sempre a partir de que valor?
- Um cronômetro precisa medir o tempo em segundos, caso algum comando esteja em outra unidade, será necessária uma conversão.
- Um cronômetro além de medir os segundos, mede também os milésimos de segundos, por exemplo: 2,42 (2 segundos e 42 centésimos), logo qual tipo de variável você deve escolher?
- Para observar os resultados, abra o *monitor serial*

#### 3.5.2 Medindo a temperatura

Usando um sensor de temperatura, meça a temperatura ambiente. No decorrer do experimento, observa temperatura e coloque o dedo em cima do sensor e observe se a temperatura sofre alguma alteração.

Para realizar esta atividade serão necessários os seguintes materiais:

- 1 LM35;
- 1 *protoboard*;
- 3 *jumper*s;
- Uma placa Arduino.

Dicas:

- Veja a seção 3.3.1 para saber como fazer a instalação correta do LM35 e a operação matemática necessária para realização. Para observar os resultados, abra o *monitor serial*.

### 3.5.3 Sensor de Luminosidade

Faça um experimento para medir a intensidade luminosa, quando estiver observando as medidas, coloque algum objeto que faça uma sombra em cima do sensor e observe o que acontece. Posteriormente, repita o processo, só que dessa vez, aproxime uma lanterna em cima do sensor e observe o que acontece.

Para realizar esta atividade serão necessários os seguintes materiais:

- 1 LDR;
- 1 resistência de 3300 ohm;
- 1 *protoboard*;
- 3 *jumper*s;
- Uma placa Arduino.

Dicas:

- Veja a seção 3.3.2 para saber como fazer a instalação correta do LDR.
- Para observar os resultados, abra o *monitor serial*.

### 3.5.4 Ligando um Motor

Ligue um motor elétrico DC usando o Arduino. Motores elétricos geralmente são utilizados para automação, pense da seguinte maneira: abrir ou fechar uma cortina, fazer um carrinho de controle remoto etc., tudo isso é possível graças a um motor dc.

Para realizar esta atividade serão necessários os seguintes materiais:

- 1 motor DC;
- 1 resistência de 100 ohm;
- 1 *protoboard*;
- 2 *jumper*s;
- Uma placa Arduino.

Dicas:

- Veja a seção 3.4.2 para ver a instalação do motor

### **3.5.5 Utilizando uma relê para ligar um LED**

Como já vimos anteriormente, a relê tem a função de uma chave, por exemplo, caso queiramos fazer um detector de presença com Arduino, podemos utilizar um sensor de luminosidade que ativa a relê fazendo com que uma lâmpada acenda de acordo com a movimentação do ambiente.

Para realizar esta atividade serão necessários os seguintes materiais:

- 1 LED;
- 1 resistor de 220 $\Omega$ ;
- 1 relê
- 1 *protoboard*
- 4 *jumper*s
- Uma placa Arduino

Dicas:

- Veja a seção 3.4.1 para ver instalação correta do LED e seção 3.4.3 para entender o funcionamento da relê.

## Capítulo IV- Projetos

Agora que você já viu a introdução à programação, as principais abas do Ardublock, divisor de tensão, os sensores e atuadores, utilize os conhecimentos adquiridos até aqui para execução dos projetos. Dividam-se em três grupos e os projetos serão determinados por sorteio. Montem o circuito inicialmente e antes de conectar a placa Arduino no computador, chame o professor para verificar se as conexões estão corretas, posteriormente, crie e execute a programação de seu projeto. Caso não obtenha êxito na primeira vez, não desista, tente novamente verificando as conexões da placa e a programação que você fez e sempre que tiver dúvidas, chame o professor.

### 4.1 Projeto I – Controlando uma Lâmpada com o Arduino

Utilize um *reed switch* (sensor) com um resistor para criar uma divisão de tensão. Se você fechar o circuito com *reed switch* aproximando com ímã, você terá um estado ALTO, logo, utilize o relê (atuador) para ligar uma lâmpada com o Arduino quando sinal estiver o estado alto, caso ao contrário, deixa a Lâmpada desligada.

#### 4.1.1 Montagem do Circuito do Projeto I

Para montagem dos circuitos serão utilizados os seguintes materiais:

- 1 *reed switch*;
- 1 relê
- 1 lâmpada
- 1 resistência de 220  $\Omega$
- 1 *protoboard*
- 4 Jumpers (macho-macho)

### 4.2 Projeto II – Ventilador Inteligente

Utilize o LM35 (sensor) que medirá a temperatura ambiente, se a temperatura superar os 25°C então o motor (atuador) irá ligar, caso contrário, ele deverá ficar desligado. Esse é o projeto do “Ventilador Inteligente”, ou seja, um ventilador que liga de acordo com uma dada temperatura. Para testar esse projeto,

utilize seu dedo em cima do LM35 para aumentar a temperatura medida pelo sensor.

#### 4.2.1 Montagem do Circuito do Projeto II

Para montagem dos circuitos serão utilizados os seguintes materiais:

- 1 LM35
- 1 resistor de  $100\Omega$
- 1 *protoboard*
- 1 motor DC
- 5 *jumpers* (macho-macho)

#### 4.3 Projeto III – Sinaleira inteligente

Utilize o LDR (sensor) com uma resistência para medir a intensidade luminosa, pegue um valor padrão médio, ou seja, quanto mais ou menos é o valor que aparece no monitor serial, depois juntamente com três LEDs (que deverão estar associados em série cada um com sua respectiva resistência), para construir uma “Sinaleira Inteligente”, ou seja, a temporização da sinaleira varia de acordo com intensidade luminosa no sensor LDR, mas o que isso significa? Quando a luminosidade for baixa, a sinaleira ficará mais tempo com o sinal fechado (com poucos carros), se luminosidade for média (número médio de carros), ficará um tempo moderado com o sinal fechado e se a intensidade luminosa for alta (com muitos carros) o sinal ficará aberto mais rápido. Para testar esse projeto, faça uma sombra com sua mão ou um caderno, para diminuir a intensidade luminosa e para aumentar a intensidade use uma lanterna ou o flash do celular.

##### 4.3.1 Montagem do Circuito do projeto III

Para montagem dos circuitos serão utilizados os seguintes materiais:

- 3 LED coloridos (verde, laranja e vermelho)
- 3 resistores de  $220\Omega$
- 1 resistores de  $3,3k\Omega$
- 1 LDR
- 9 *Jumpers* com ambas extremidades macho
- 1 Placa Arduino
- 1 *Protoboard*

#### **4.4 Modificação dos Projetos**

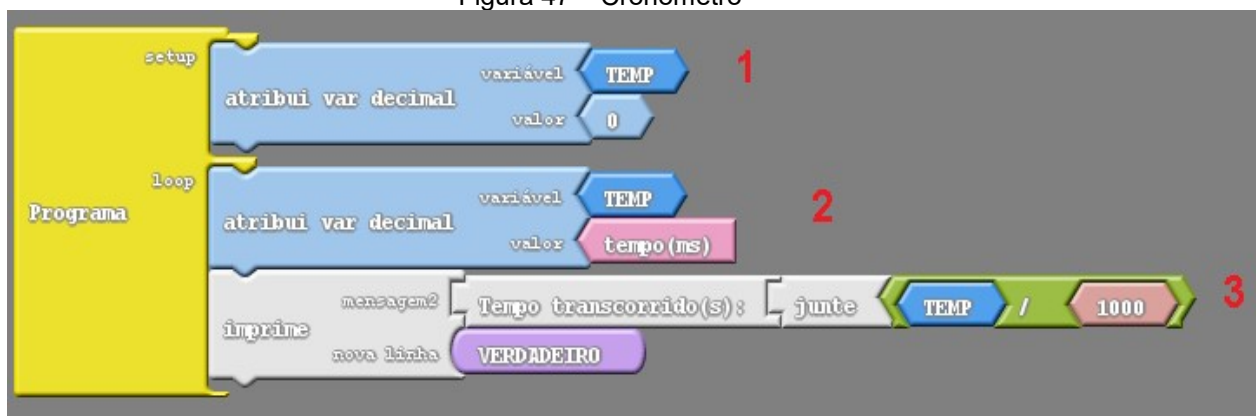
Após a realização dos projetos, realize pequenas modificações nos projetos executados pelos outros grupos, adicionando um sensor ou um atuador e modificando também sua programação. Discuta com os seus colegas qual modificações você decidiu fazer e motivo por trás dessa modificação.

## Soluções:

### Problema 3.5.1 Cronômetro

Observe o programa de um cronômetro na Figura 40, mesmo que não fique claro alguns comandos abaixo, fiquem tranquilos que iremos tratar com mais cuidado posteriormente:

Figura 47 – Cronômetro



Fonte: Autor

O nosso programa acima foi dividido em 3 comandos principais. O comando escolhido para iniciar a nossa programação foi o comando *Programa*.

**Comando 1:** Este bloco está dentro do *Setup* e estamos atribuindo um valor inicial e um nome para uma variável decimal (número com virgula), o valor escolhido foi igual a zero e o nome dessa variável é *TEMP*.

**Comando 2:** Agora estamos utilizando novamente o comando de atribuir um valor para uma variável decimal, a diferença é que estamos usando o comando *tempo(ms)* para atribuir ao valor de nossa variável decimal.

**Comando 3:** O terceiro e último comando que estamos utilizando é para imprimir uma mensagem com o valor de nosso tempo, a mensagem será “*Tempo transcorrido (s):*” e para juntar o valor do tempo com a mensagem utilizamos o comando *Junte*. Como nossa variável *TEMP* está em milissegundos dividimos o valor dela por 1000 para encontrar o valor em segundos. Ainda neste comando,

marque *VERDEIRO* para nova linha, isto vai fazer com que cada valor fique em uma linha diferente. Para visualizar as medidas clique em *Monitor Serial*, no Ardublock. Lembrando que antes o programa deve ser enviado para a interface Arduino IDE e, posteriormente, carregado na placa Arduino.

Observação: Apesar do comando *tempo(ms)* ser uma variável do tipo inteira, foi utilizado uma variável do tipo decimal para medida de tempo transcorrido devido ao fato de que a variável decimal nos retorna um valor com precisão de 2 casas decimais (Figura 48.a), enquanto a variável inteira só nos retorna um valor inteiro (Figura 48.b).

Na Figura 48.a valores do variável tempo em segundos, usando variável do tipo inteira. Na Figura 48.b valores do variável tempo em segundos, usando variável do tipo decimal.



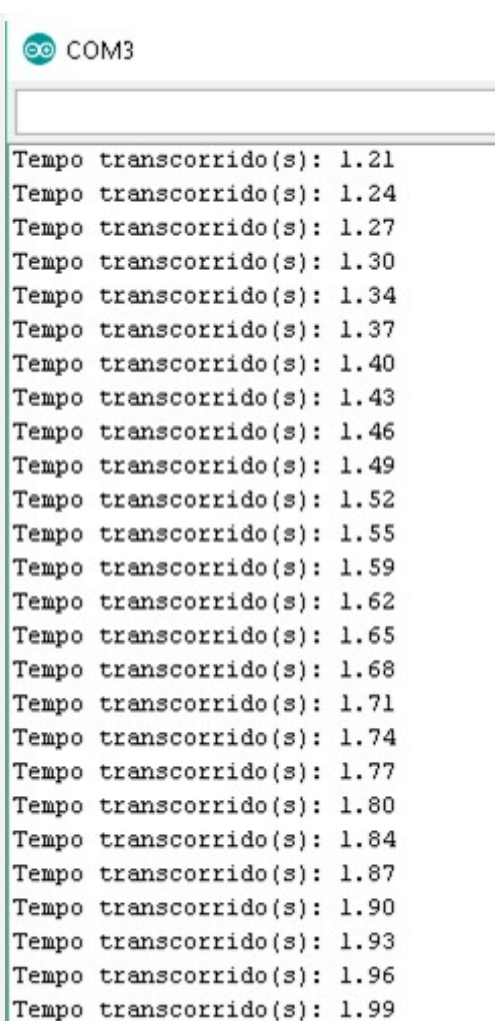
**Figura 48.a – Cronometro no Monitor Serial (variável inteira)**



```
COM5 (Arduino/Genuino Uno)
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 0
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
Tempo transcorrido(s): 1
```

Fonte: Autor

**Figura 48.b – Cronometro no Monitor Serial (variável decimal)**



```
COM3
Tempo transcorrido(s): 1.21
Tempo transcorrido(s): 1.24
Tempo transcorrido(s): 1.27
Tempo transcorrido(s): 1.30
Tempo transcorrido(s): 1.34
Tempo transcorrido(s): 1.37
Tempo transcorrido(s): 1.40
Tempo transcorrido(s): 1.43
Tempo transcorrido(s): 1.46
Tempo transcorrido(s): 1.49
Tempo transcorrido(s): 1.52
Tempo transcorrido(s): 1.55
Tempo transcorrido(s): 1.59
Tempo transcorrido(s): 1.62
Tempo transcorrido(s): 1.65
Tempo transcorrido(s): 1.68
Tempo transcorrido(s): 1.71
Tempo transcorrido(s): 1.74
Tempo transcorrido(s): 1.77
Tempo transcorrido(s): 1.80
Tempo transcorrido(s): 1.84
Tempo transcorrido(s): 1.87
Tempo transcorrido(s): 1.90
Tempo transcorrido(s): 1.93
Tempo transcorrido(s): 1.96
Tempo transcorrido(s): 1.99
```

Fonte: Autor

### Problema 3.5.2 Medindo a temperatura

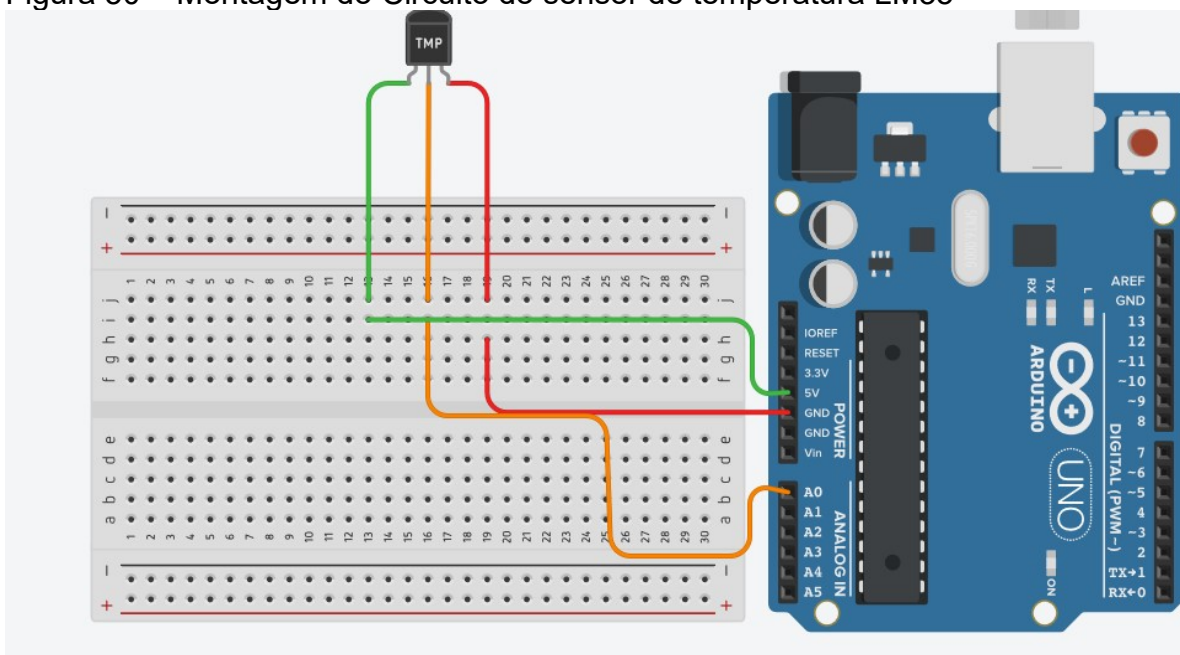
Figura 49 – Código da medida de temperatura com LM35



Fonte: Autor

A montagem do circuito deve se assemelhar a:

Figura 50 – Montagem do Circuito do sensor de temperatura LM35



Fonte: Produzido pelo autor no TinkeCard

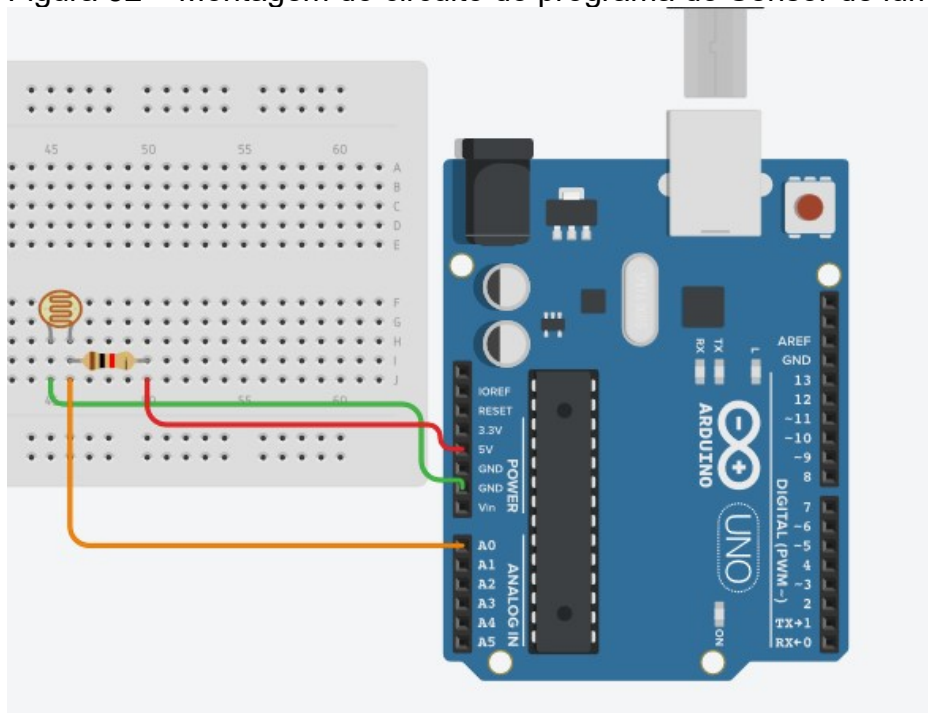
### Problema 3.5.3- Sensor de Luminosidade

Figura 51 – Código do código Sensor de Luminosidade



Fonte: Autor

Figura 52 – Montagem do circuito do programa do Sensor de luminosidade



Fonte: Autor

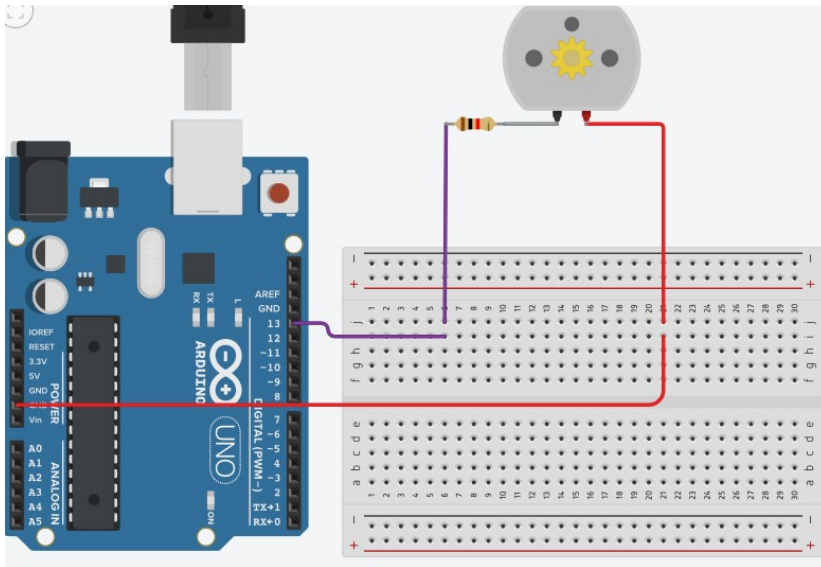
### Problema 3.5.4- Ligando um Motor

Figura 53 – Código para ligar um motor no Ardublock



Fonte: Autor

Figura 54 – Montagem de um circuito de ligar um motor



Fonte: Autor

## 5 REFERÊNCIAS

RIBEIRO, J.D. **Explorando as possibilidades de inserção da plataforma Arduino no Ensino de Ciências da Educação Básica**. 2015. 168 f. Dissertação de Mestrado. Curso de Pós-graduação em Ensino de Ciências, Universidade Federal do Pampa, Bagé, 2017.

<<http://arduino.cc/en/Reference/HomePage>> acesso em 29 de abril de 2019

.

<<http://playground.arduino.cc/Portugues/HomePage>> acesso em 29 de abril de 2019.

## GUIA DE INSTALAÇÃO ARDUBLOCK

**Guia de Instalação do Ardublock para o Arduino ide 1.82 ou superior**  
Este guia foi retirado do site <http://teachduino.ufsc.br/2017/09/22/novo-ardublock-edicao-para-fisica/> na data de 3 de maio de 2019 e adaptado pelo autor da apostila.

### **Instalação:**

- O primeiro passo é baixar o Ardublock no link abaixo:

<http://teachduino.ufsc.br/2017/09/22/novo-ardublock-edicao-para-fisica/>

OBS: Para instalar o Ardublock o programa do Arduino deve estar fechado:

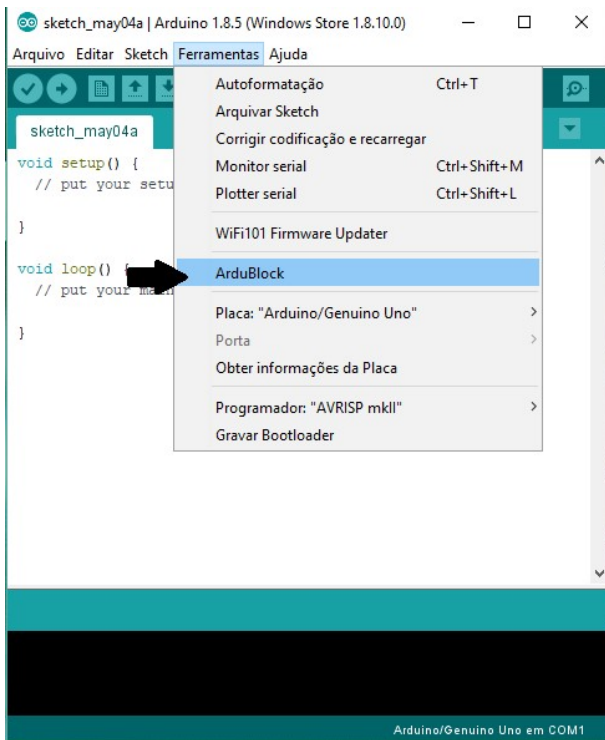
- Vá na pasta “Meus Documentos” ou em “C:/users/seu\_usuario/Documents/”
- Abra a pasta escrito “Arduino” e crie uma nova pasta chamada “tools”(a pasta deve estar com o nome todo em minúsculo)
- Posteriormente, abra a pasta “tools” e crie a pasta “ArduBlockTool”(Exatamente como está escrita). O próximo passo é abrir a pasta “ArduBlockTool” e criar uma pasta chama “tool”(tudo em minúsculo).
- O ultimo passo de instalação é copiar o arquivo baixo no link para dentro da pasta “tool”.

OBS: É necessário ter o java instalado, entretanto, se o programa Arduino IDE roda perfeitamente, significa que o programa já está instalado

### **Testar o Ardublock:**

Abra o Arduino IDE vá em “Ferramentas” e selecione a opção “ArduBlockTool”, conforme a imagem abai

Figura 1 – Aba Ardublock



Fonte: Autor

Fazendo programa teste (blink):

1. Vá em “controle” e selecione o código “sempre”
2. Vá em pino e selecione “seta pino digital”, selecionar a porta 13 e colocar no estado “Alto” e coloque dentro do bloco “sempre”.
3. Vá em “controle” e selecione o código “delay MILLIS(milissegundos)” estabeleça o valor 1000 e coloque dentro do bloco sempre.
4. Vá em pino e selecione “seta pino digital”, selecionar a porta 13 e colocar no estado “Baixo” e coloque dentro do bloco “sempre”.
5. Vá em “controle” e selecione o código “delay MILLIS(milissegundos)” estabeleça o valor 1000 e coloque dentro do bloco sempre.

A figura abaixo mostrará a estrutura do programa:

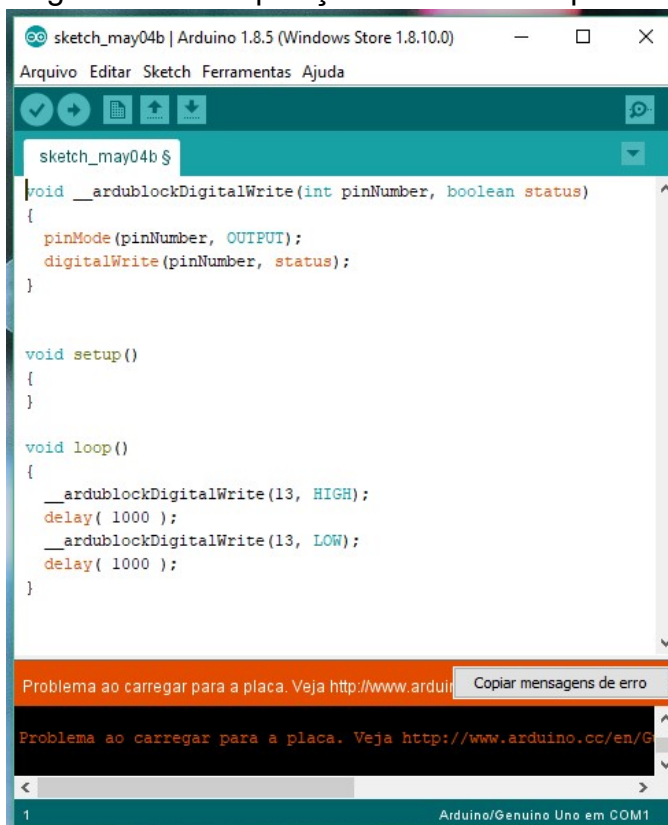
Figura 2 – Código do Programa



Fonte: Autor

Salve o programa na aba “Salvar” e logo após clique em “Enviar para Arduino”, se o código foi transcrito para o Arduino IDE, o programa ArduBlock funcionou sem problemas, basta compilar e executar placa Arduino.

Figura 3 – Transposição do Ardublock pro Arduino IDE



The image shows a screenshot of the Arduino IDE interface. The window title is "sketch\_may04b | Arduino 1.8.5 (Windows Store 1.8.10.0)". The menu bar includes "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". The toolbar contains icons for saving, running, and uploading. The sketch editor shows the following code:

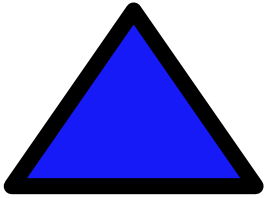

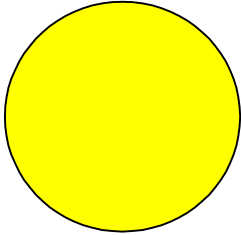


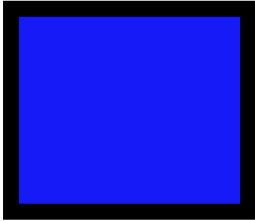

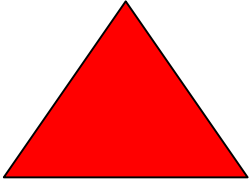

```
sketch_may04b $  
  
void __ardublockDigitalWrite(int pinNumber, boolean status)  
{  
  pinMode(pinNumber, OUTPUT);  
  digitalWrite(pinNumber, status);  
}  
  
void setup()  
{  
}  
  
void loop()  
{  
  __ardublockDigitalWrite(13, HIGH);  
  delay( 1000 );  
  __ardublockDigitalWrite(13, LOW);  
  delay( 1000 );  
}
```

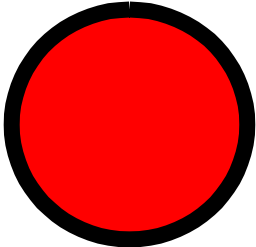
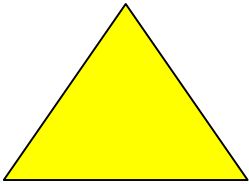

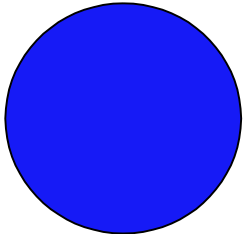





Below the code editor, an orange error message box is visible: "Problema ao carregar para a placa. Veja <http://www.arduino.cc/en/Guide/Tools> para obter mais informações." A button labeled "Copiar mensagens de erro" is also present. The status bar at the bottom indicates "1" and "Arduino/Genuino Uno em COM1".

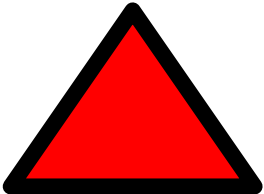

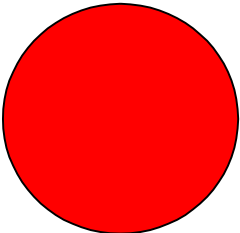
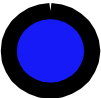
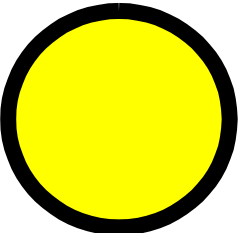
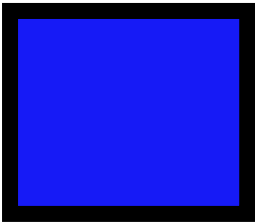

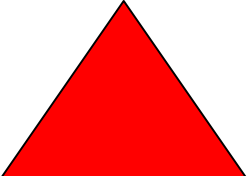

Fonte: Autor


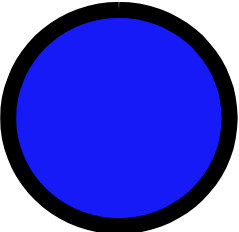




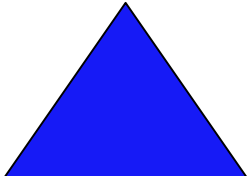




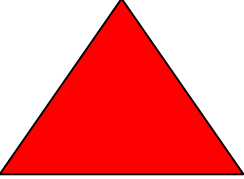
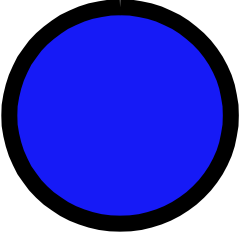

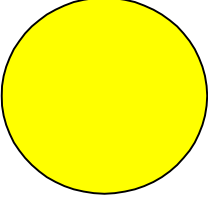



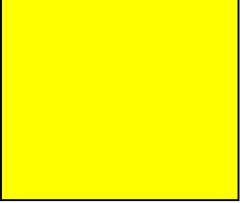

## CARTERLAS BINGO LÓGICO

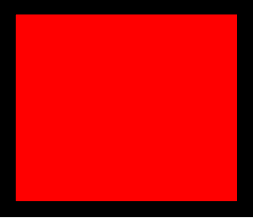

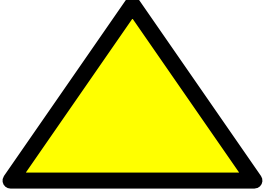
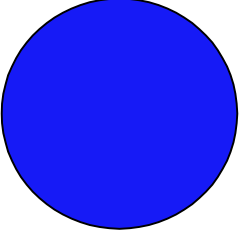



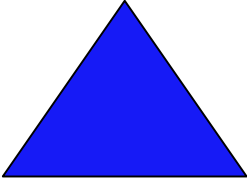

|  |  |  |
|--|--|--|
|   |   |   |
|   |   |   |
|  |  |  |

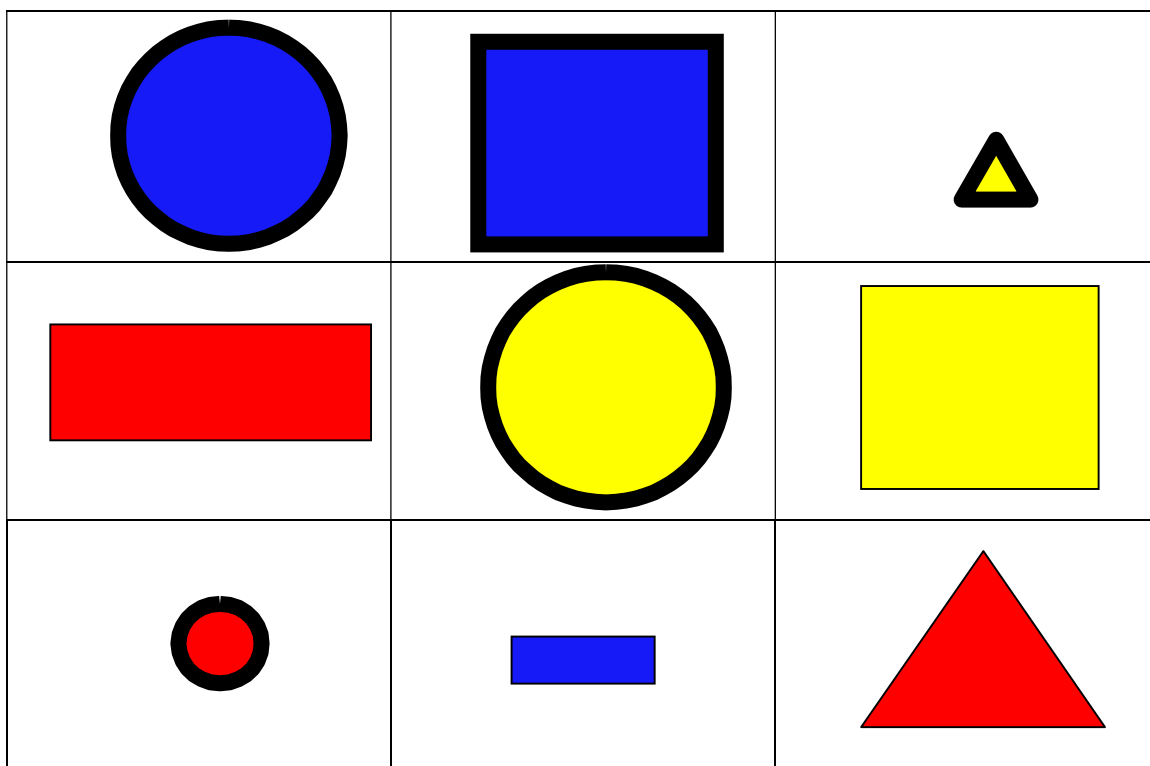
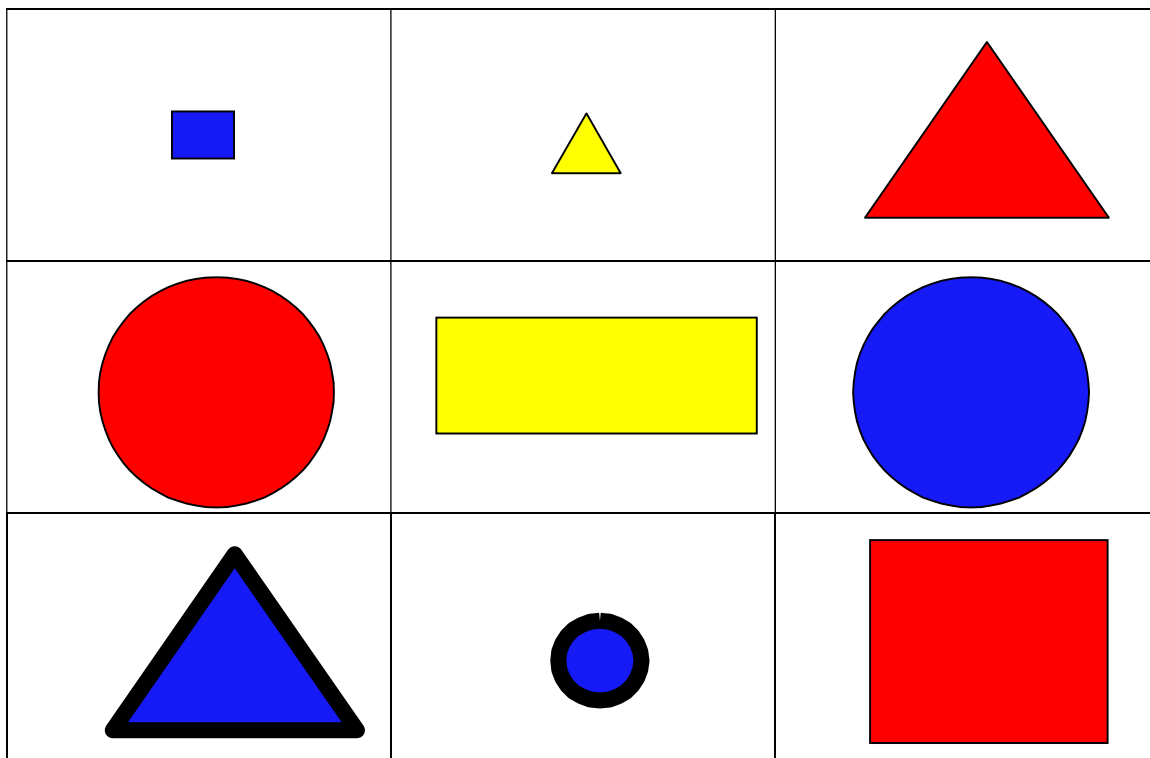
|   |   |   |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |

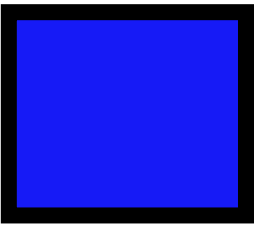




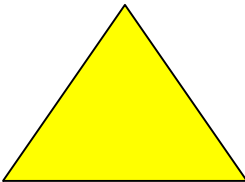
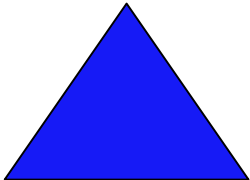

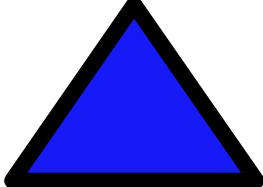
|   |  |  |
|---|--|--|
|  |   |   |
|  |   |   |
|  |  |  |


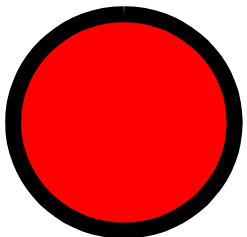

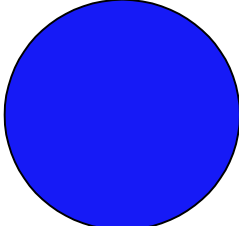


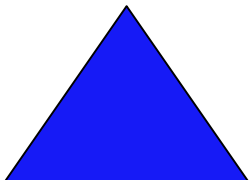


|   |   |   |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |

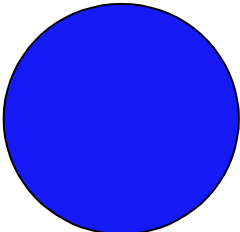
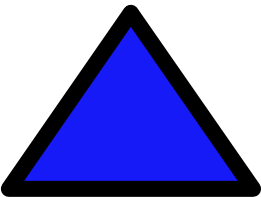





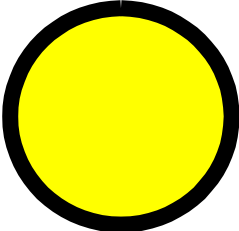

|  |  |   |
|--|--|---|
|   |   |  |
|   |   |  |
|  |  |  |

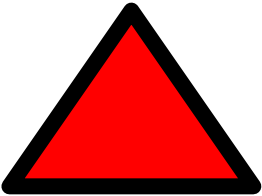
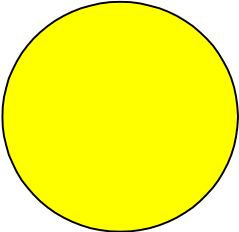
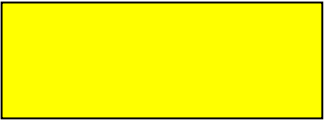


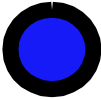

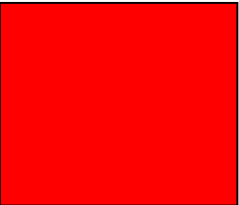

|   |   |   |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |


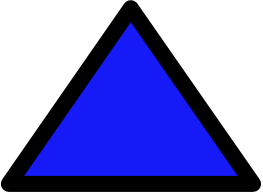
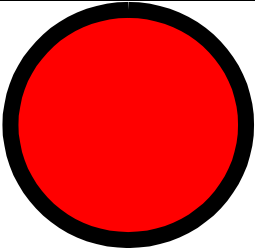

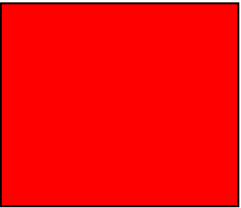
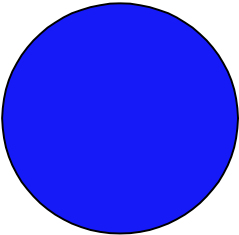


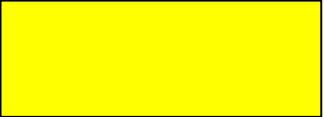



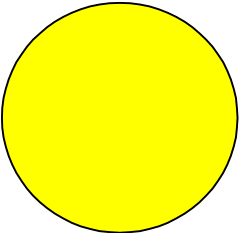

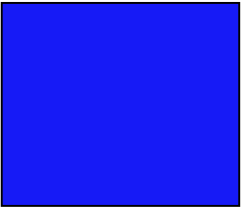
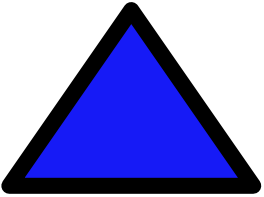

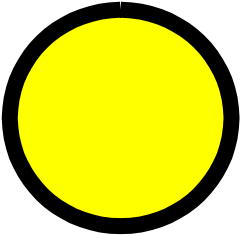

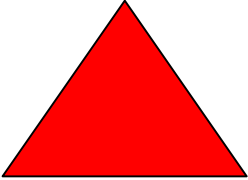
|  |   |  |
|--|---|--|
|   |  |   |
|   |  |   |
|  |  |  |


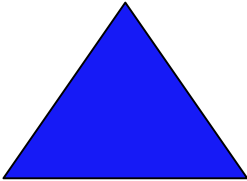


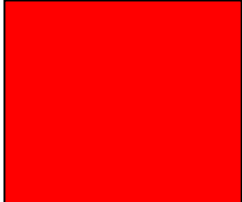

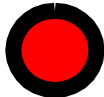


|   |   |   |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
|--|--|--|
|   |   |   |
|   |   |   |
|  |  |  |

|   |   |   |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |

|  |   |  |
|--|---|--|
|   |  |   |
|   |  |   |
|  |  |  |

|   |   |   |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |

|   |   |   |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |