

2º ENCONTRO

Introdução à Lógica de Programação

Introdução

A lógica está presente em nossa vida sempre que pensamos, falamos e escrevemos, pois para realizar essas ações necessitamos que os pensamentos estejam ordenados de modo a alcançar o resultado esperado. Na verdade a lógica consiste simplesmente na organização e explicação de um pensamento.

Poderemos identificar a utilização da lógica no nosso dia-a-dia quando analisamos situações como:

Todo mamífero bebe leite;
O homem bebe leite. ➔ Por tanto concluísse que:
O homem é um mamífero.

Analizando estes argumentos em uma sequencia lógica podemos chegar a uma conclusão, como demonstrado acima.

(Retirado do livro *Lógica de programação*, de Camila Ceccatto da Silva e Everaldo Antonio de Paula, adaptado)



Linguagem de Programação

Podemos imaginar o computador como uma super calculadora capaz de fazer cálculos muito mais rápido que nós, mas para isso devemos dizer para o computador o que deve ser calculado e como deve ser calculado. A função das linguagens de programação é exatamente essa, ou seja, servir de um meio de **comunicação** entre computadores e humanos, assim como os humanos fazem com os humanos: usam

da linguagem para poder conversar.

Quando programamos em uma linguagem de programação primeiramente criamos um arquivo de texto comum contendo a lógica do que deve ser feito, ou seja, é onde falamos ao computador,

The screenshot shows a window titled "ballaccel.kbs - BASIC-256". The menu bar includes File, Edit, View, Run, and About. The toolbar has New, Open, Save, Run, Debug, Step, Stop, Cut, Copy, and Paste. The code area contains the following BASIC code:

```

loop1:
rem clg
rem change the downward velocity according to the
acceleration
yvel = yvel + yacc
y = y + yvel
x = x + xvel
if y > 289 then yvel = -0.9 * yvel : y = 289 :
xvel = xvel * 0.9
if x > 285 then xvel = -xvel : x = 285
if x < 10 then xvel = -xvel : x = 10
color darkblue
rem rect 0,0,300,300
gosub drawBall
if xvel * xvel < 0.0001 then end
goto loop1

drawBall:
color darkgray
circle x, y, r
color gray
circle x, y, r - 2
refresh
return

```

The "Text Output" panel says "Using Newton's laws to make the ball bounce realistically". The "Graphics Output" panel shows a ball bouncing on a blue background.

o que queremos. Este arquivo de texto é chamado de **programa**, cada palavra de ordem dentro do programa é chamada de instrução. Após criarmos o programa, este deve ser traduzido para linguagem entendida pelo computador a partir da linguagem que estamos utilizando. Um segundo arquivo que chamamos de executável ou programa, será gerado e interpretado diretamente pelo computador.

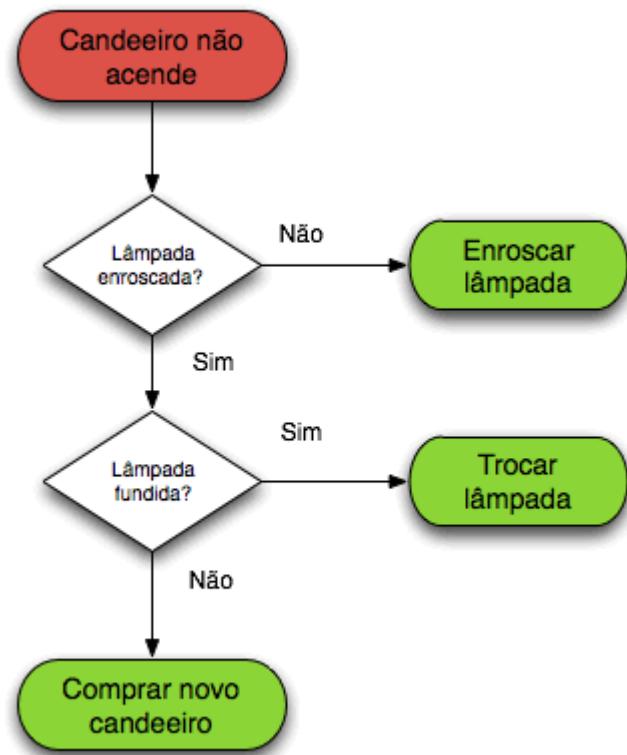
(Retirado de www.infoescola.com, adaptado)

Algoritmo

Algoritmo é uma sequencia lógica de passos que levam a um determinado objetivo. Apesar de a palavra algoritmo parecer estranha e desconhecida, é claro que todos nós sabemos construir algoritmos.

Se isto não fosse verdade, não conseguiríamos sair de casa pela manhã, ir à escola, decidir qual o melhor caminho para chegar a um lugar, voltar para casa, etc. Para que tudo isto seja feito é necessário uma série de entradas do tipo: a que hora acordar, a que horas sair de casa, qual o melhor meio de transporte, etc.

Um fator importante é que pode haver mais de um algoritmo para resolver um problema. Por exemplo, para ir de casa até o colégio, podemos escolher diversos meios de transportes em função do preço, conforto, rapidez, etc. A escolha será feita em função do critério que melhor se adequar as nossas necessidades.



(Retirado do livro **Lógica de programação**, de Camila Ceccatto da Silva e Everaldo Antonio de Paula, adaptado).

Veja e analise a solução necessária para o exemplo abaixo:

Escapando de um Labirinto

Problema	 Robô
	Instruções Possíveis
Solução	<p>Encontrar uma sequência de instruções possíveis para que o robô seja capaz de sair do labirinto</p>

Algoritmo

Algoritmo é uma sequência lógica de passos que levam a uma solução. Fazemos o uso de algoritmos nas nossas ações diárias como, os passos que realizamos para tomar banho. O algoritmo deve ter início, instruções e fim.

Retirado do Curso de Programação Computadores, Prof. Aníbal Tavares e Profa. Cassilda Ribeiro, disponível em <http://www.feg.unesp.br/~cassilda/aulas_PC/C_Aulas2.pdf>

Os algoritmos, portanto, podem ser divididos em partes, que são elas:

Início;

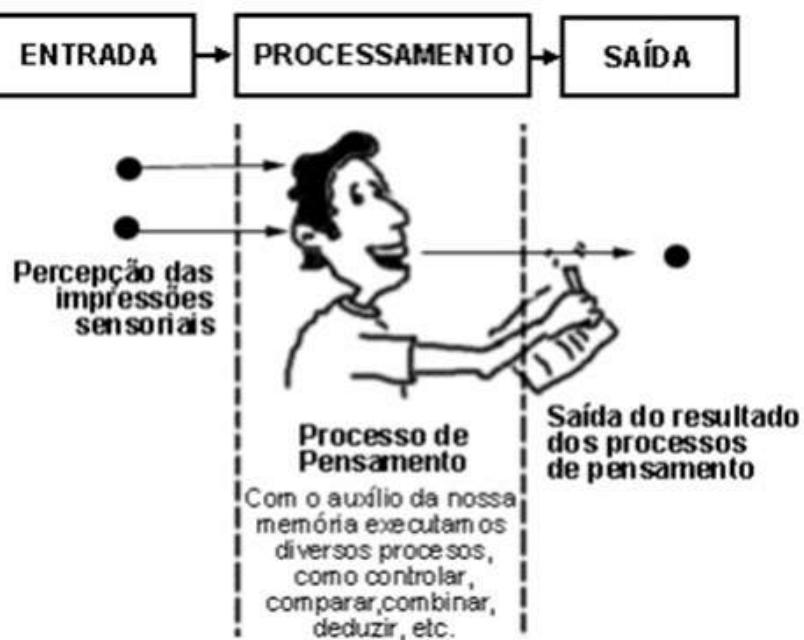
Entrada: dados que devem ser lidos, ou fornecidos pelo usuário;

Processamento: onde acontecem os cálculos em si, os dados de entrada são manipulados a fim de se gerar uma resposta (um resultado);

Saída: são os resultados de seu processamento.;

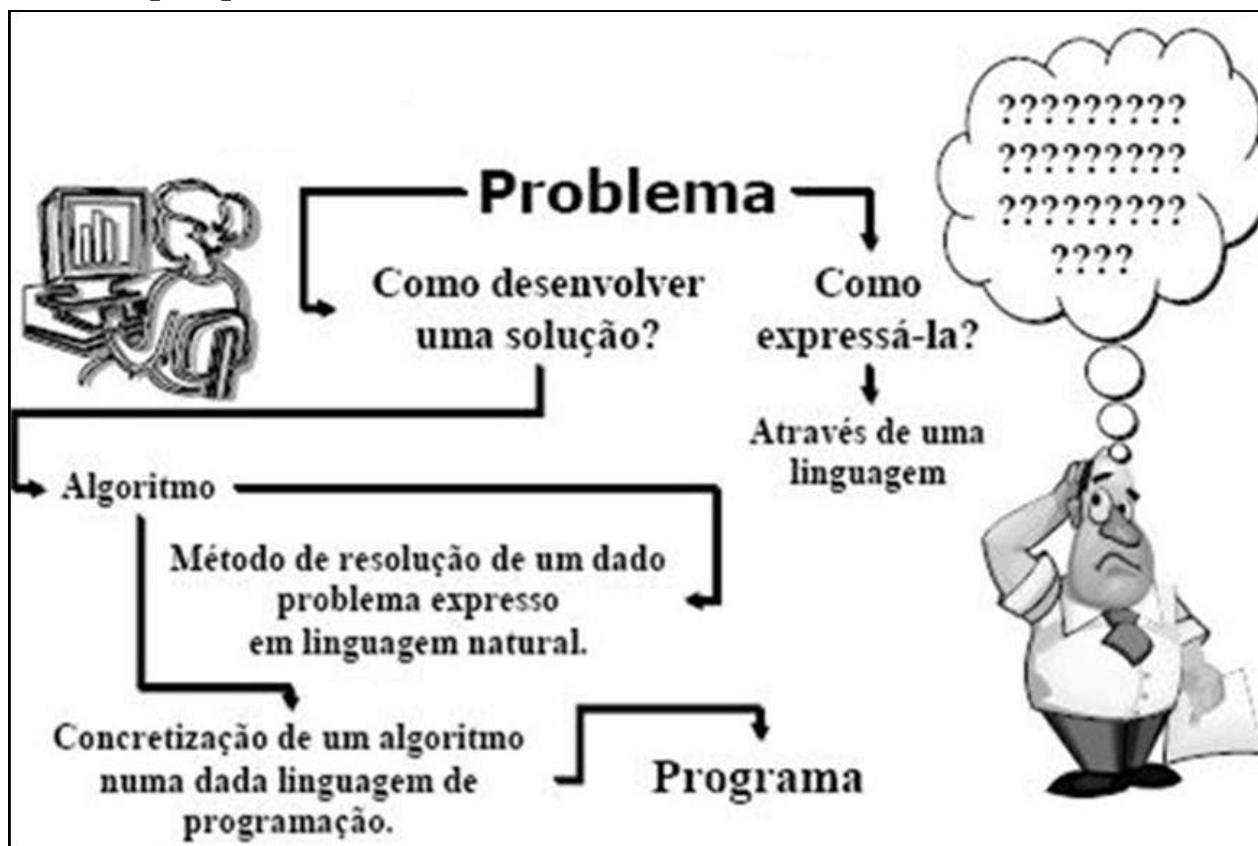
Fim.

Analogia com o homem



A Lógica e a Informática

Para o desenvolvimento de programas de computador é necessário o desenvolvimento de uma lógica bem estruturada para que o computador possa realizar com sucesso os procedimentos desejados para a solução do problema proposto. Portanto, a partir de um problema proposto, temos:



Como exemplo, iremos usar uma receita que é um algoritmo descrito em linguagem natural. Passos para o desenvolvimento de uma lógica de programação:

- 1 - Analise o que deve ser realizado;
- 2 - Imagine como o seu cérebro processa esta ação;
- 3 - Divida isto em passos distintos e muito específicos;
- 4 - Estruture estes passos em uma sequência lógica;
- 5 - Verifique o funcionamento e exatidão da sua lógica.

Podemos desenvolver a lógica de simples atividades de nosso cotidiano, como “fritar um ovo”:

- 1- É necessário fritar um ovo;
- 2- Para fritar um ovo deve-se ir até a cozinha, ter um ovo, uma frigideira, óleo, um fogão e gás. O fogão deve ser ligado, a frigideira aquecida, e o óleo aquecido na frigideira, posteriormente o ovo deve ser frito no óleo quente;

Passo 1 - Adquirir:

- 1- Obtenha um ovo, caso não tenha obtido um ovo, obtenha um ovo, faça isso até obter um ovo;
- 2- Obtenha uma frigideira, caso não tenha obtido uma frigideira, obtenha uma frigideira, faça isso até obter uma frigideira;
- 3- Obtenha o óleo, caso não tenha obtido o óleo, obtenha o óleo, faça isso até obter o óleo;
- 4- Obtenha um fogão, caso não tenha obtido um fogão, obtenha um fogão, faça isso até obter um fogão;
- 5- Obtenha o gás, caso não tenha obtido o gás, obtenha o gás, faça isso até obter o gás;

Passo 2 - Preparar:

- 1- Ligue o fogo, caso o fogo não tenha ligado, ligue o fogo,
- 2- faça isso até ligar o fogo;
- 3- Coloque a frigideira no fogo;
- 4- Aqueça a frigideira, caso a frigideira não esteja aquecida, aqueça a frigideira, faça isso até a frigideira aquecer;
- 5- Coloque o óleo na frigideira;
- 6- Aqueça o óleo, caso o óleo não esteja aquecido, aqueça o óleo, faça isso até o óleo aquecer;
- 7- Quebre a casca do ovo;
- 8- Retire o ovo, se o ovo não estiver bom obtenha outro ovo, quebre a casca do ovo e retire o ovo.

Passo 3 - Cozinhar:

- 1- Coloque o ovo no óleo que está na frigideira;
 - 2- Verifique se o ovo está pronto, se o ovo está pronto, se não, verifique se o ovo está pronto, faça isso até o ovo ficar pronto;
 - 3- Retire o ovo da frigideira;

Passo 4 - Consumir:

- 1- Coloque o ovo em um prato;
 - 2- Coma o ovo.



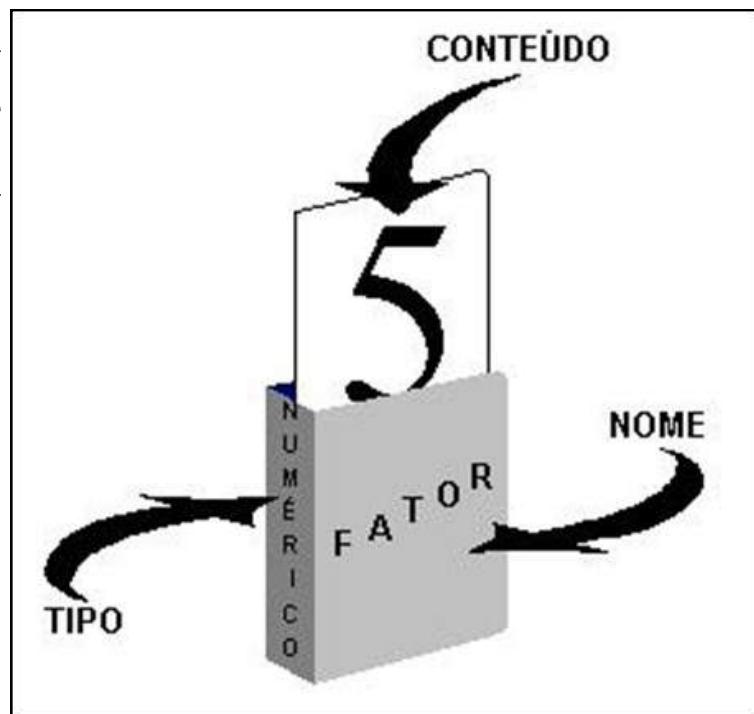
Variável

Quase todos os algoritmos que apresentamos até este ponto foram escritos em linguagem natural, ou seja, no nosso caso, habitantes do Brasil, o português.

O algoritmo em linguagem humana deve ser traduzido em uma linguagem de programação de computadores. Depois que é gerado um programa, a linguagem é outra vez traduzida para a linguagem de máquina (computador, no nosso caso) para que o computador possa entender as instruções fornecidas.

Os algoritmos irão obter os dados necessários para que a máquina possa manipular estes dados, que normalmente são fornecidos pelos usuários, e entregar resultados para os mesmos. Uma pergunta importante neste momento é: **como poderemos manipular estes dados?** A resposta é que podemos manipulá-los por meio das **variáveis**.

Imagine uma variável como uma caixinha que pode ser nomeada e esta pode guardar um valor, como no desenho ao lado.



Portanto, a variável é um recurso que pode servir para explorar uma infinidade de valores em uma situação em que há medidas desconhecidas assim como na matemática. Embora uma mesma variável possa assumir diferentes valores, ela só pode armazenar um valor a cada instante.

Existem três tipos básicos de dados que iremos manipular nos algoritmos que iremos criar:

Dados como números - **numéricos**;
 Dados como letras - **literária**;
 Dados lógicos.

Num programa, em cada instante, cada variável possui seu tipo, seu nome e seu conteúdo como exemplificado na figura, onde a variável chamada “NOME” que é do tipo “LITERÁRIA” tem como conteúdo a palavra “Maria” e a variável nomeada “IDADE” que é do tipo “NUMÉRICA” possui o número “26” armazenado enquanto as outras variáveis não tem nem conteúdo nem tipo definido:

1	2	3
Maria		
NOME	Caixa 2	Caixa 3
4	5	6
Caixa 4	Caixa 5	Caixa 6
7	8	26
Caixa 7	Caixa 8	IDADE

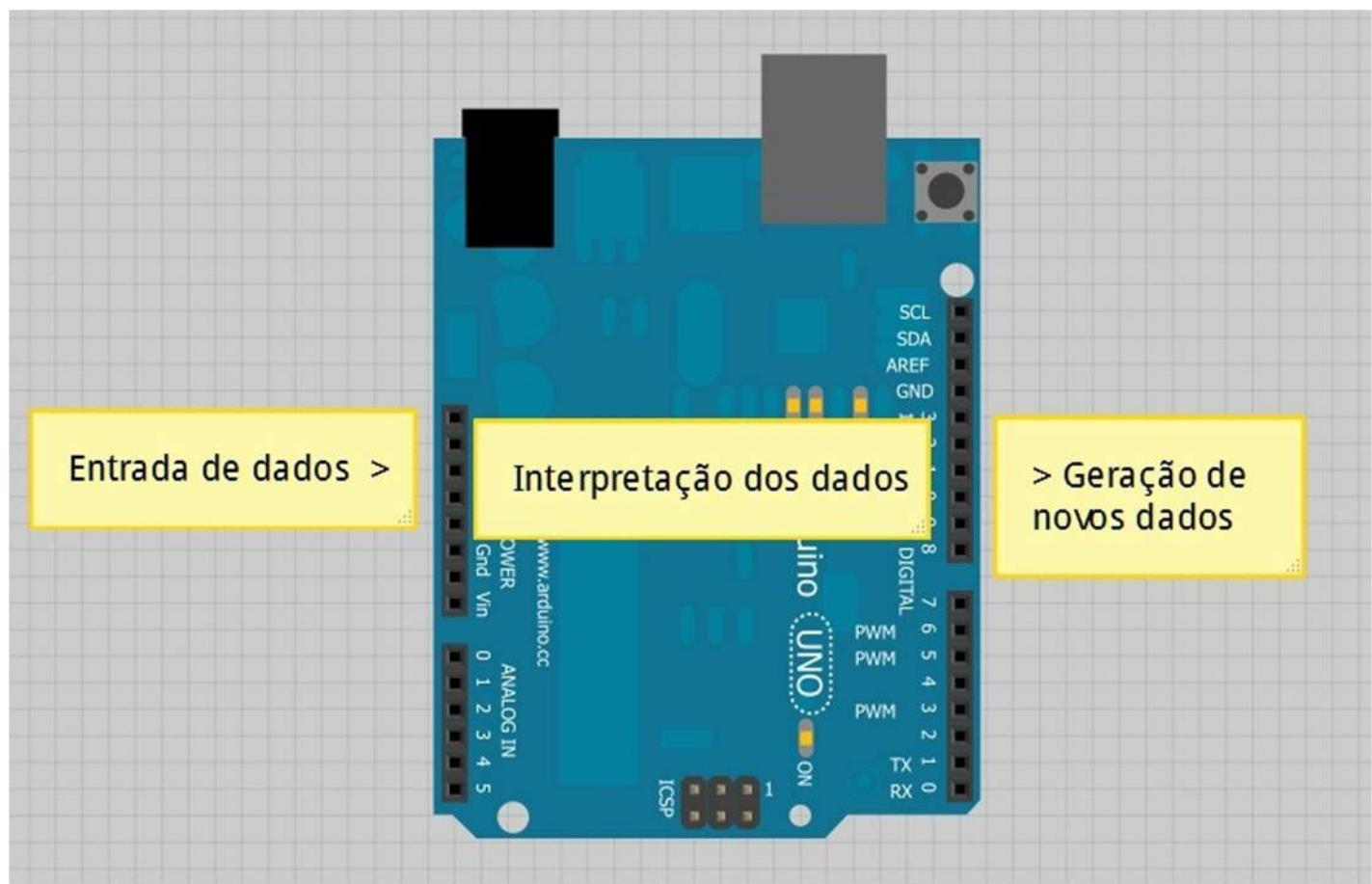
Arduino

É uma plataforma de desenvolvimento de hardware, microcontrolada de código aberto.

Em termos práticos o Arduino é um pequeno computador em que é possível se interagir com o ambiente.

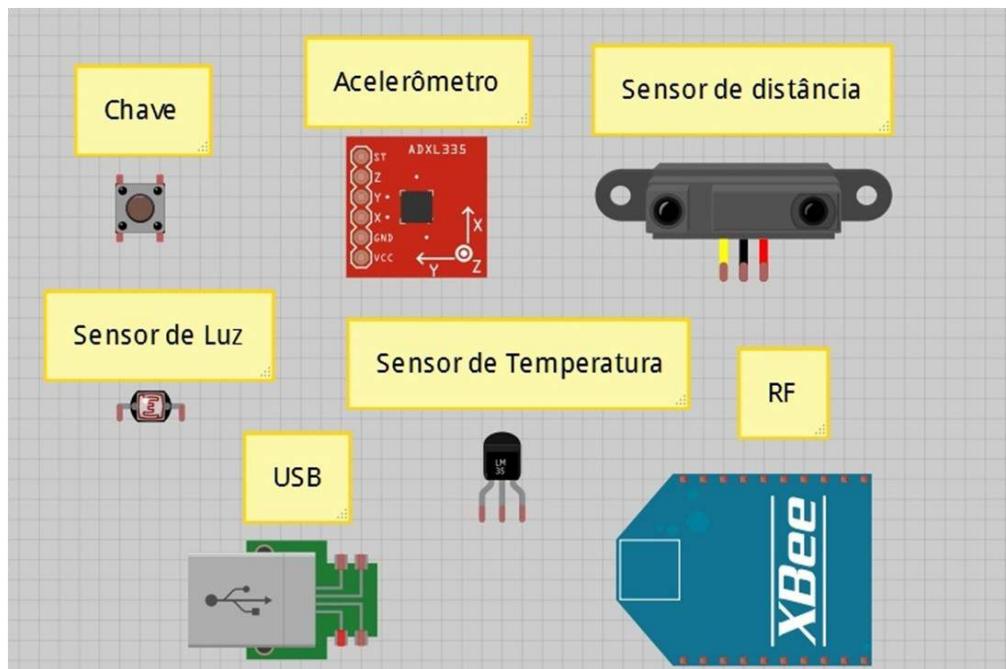


Estrutura básica e princípio de funcionamento:

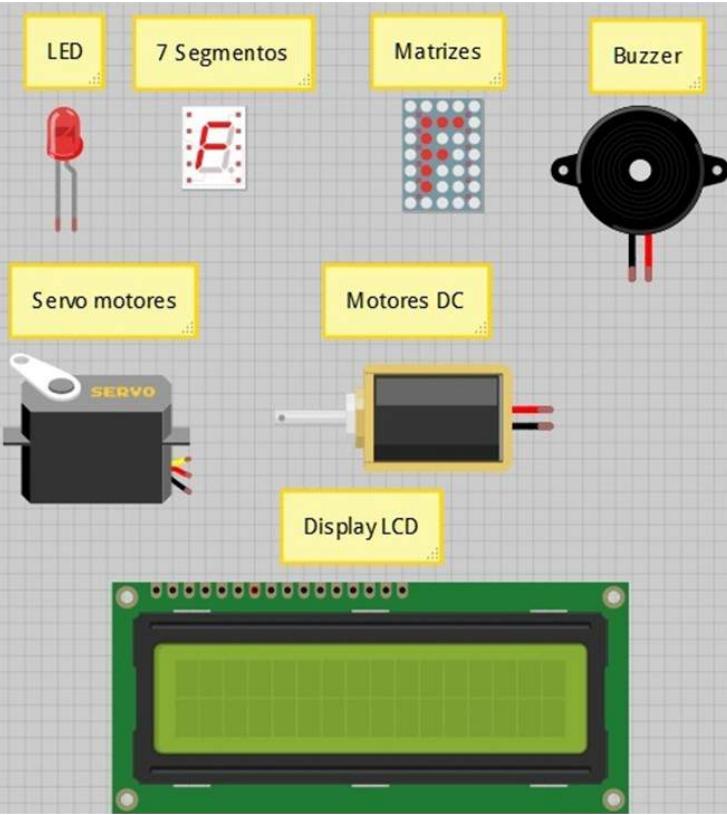


Entrada de dados

Como entrada de dados, podemos ter os dispositivos demonstrados ao lado.



Saída de dados



Como saída de dados, podemos ter os dispositivos demonstrados ao lado.

Alguns conceitos prévios

Antes de começarmos a trabalhar com o Arduino precisamos aprender alguns conceitos de eletrônica e programação que serão importantes ao decorrer do curso.



LED – (Diodo Emissor de Luz) – Pode-se compreender o LED como uma pequena lâmpada que consome baixa energia, mas que fornece uma luminosidade comparável a uma lâmpada do mesmo porte.

Push-button - (Botão ou chave) – Uma chave, ou se preferir chamar de botão, tem o mesmo princípio de funcionamento de um interruptor (liga, desliga), quando ele é pressionado o seus contatos entram em curto.



Buzzer – Emite som quando é acionado.

LDR – O LDR é um sensor de luz, que varia sua resistência elétrica de acordo com a intensidade luminosa sobre o mesmo.



Sensor de temperatura – Segue a mesma ideia do LDR, mas neste caso sua resistência varia de acordo com a temperatura e não com a luminosidade.

LCD – Display que pode ser utilizado para exibir mensagens através do Arduino.



Como funciona o Arduino?

Podemos dividir o funcionamento dele em 3 partes, leitura de dados (normalmente sensores), processamento de dados (decisões a serem tomadas a partir da leitura de dados), e atuação (após o Arduino ler os sinais e interpreta-los, ele decide o que vai fazer). Um exemplo desse funcionamento seria acender um LED com uma chave ligada ao Arduino, a chave “diz” pro Arduino que quer ligar o LED, ele por sua vez verifica se essa chave está autorizada a liga-lo (isso é feito através da programação), caso possa, o próprio Arduino se encarrega de acender o LED.

O Programa

Antes de colocarmos a mão na massa vamos entender como funciona o programa que iremos utilizar para programar o Arduino.

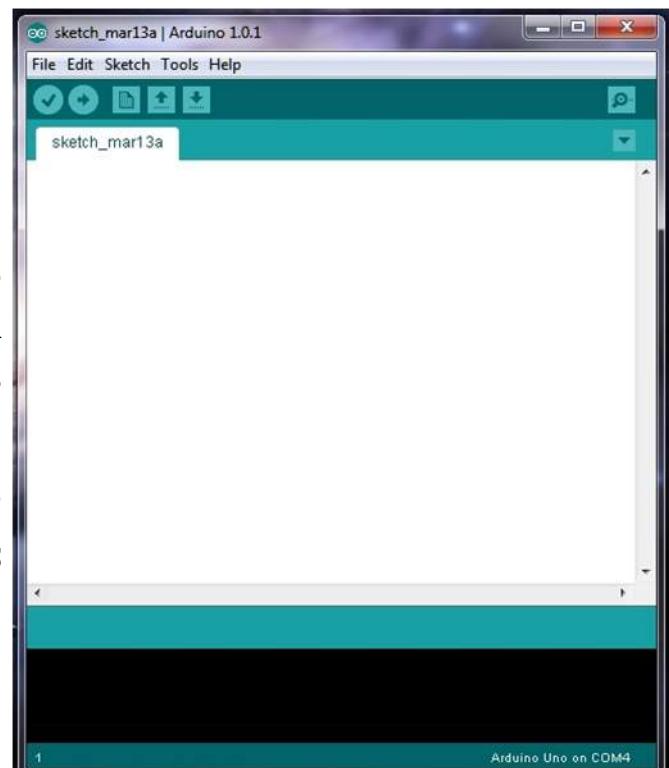
A área em branco será utilizada para colocarmos nosso código de programação. A cima delas há alguns ícones, dentre eles iremos utilizar 2, estes são:



Verifica se o programa contém erros de digitação.



Grava o programa no Arduino.



Depois de terminado o código, pressionamos o respectivo botão para verificar se digitamos alguma coisa errada (observação: esse botão não verifica erros de lógica, somente de escrita), se não houver erros já podemos gravar a programação no Arduino.

1º Experimento

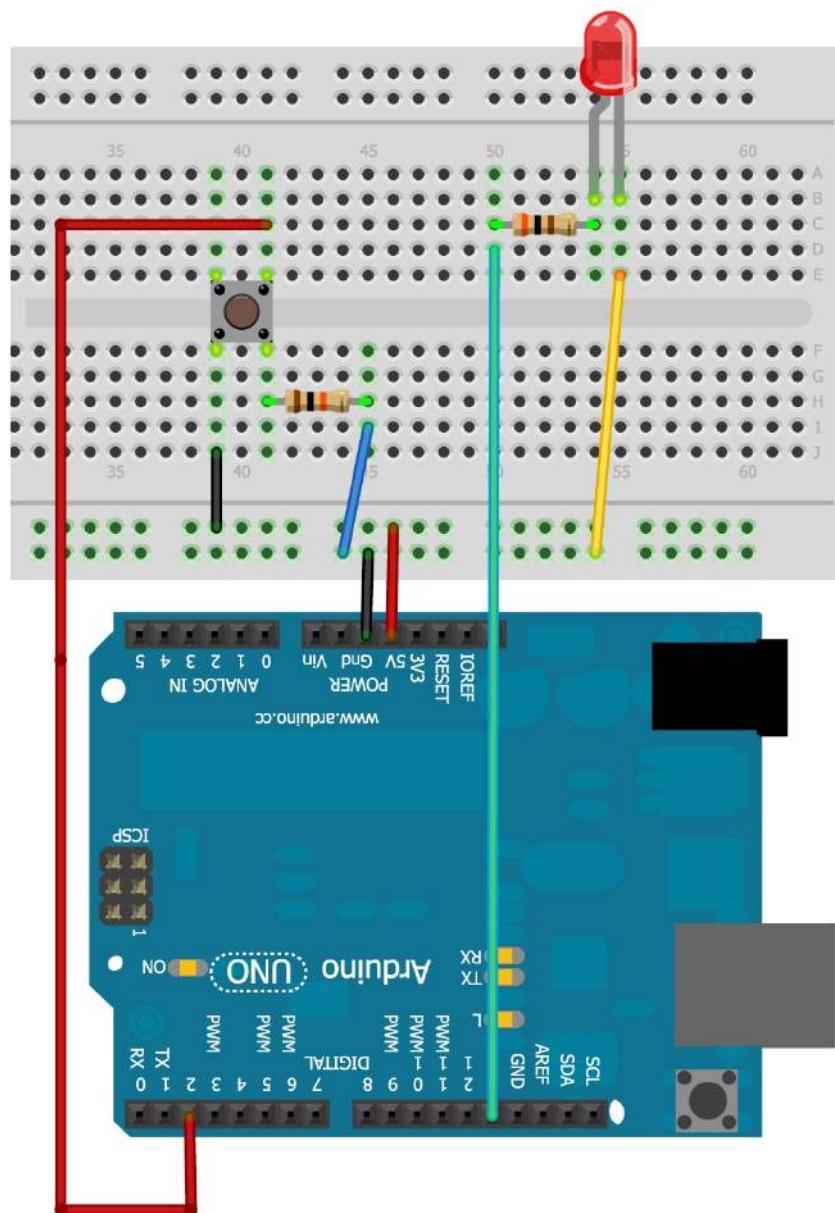
Objetivo: Acender um LED com uma push button.

Funcionamento: Ao se pressionar a push Button o LED deverá acender, e caso ela seja solta o LED deverá apagar.

Procedimentos:

(1) Monte o circuito no protoboard da maneira que segue abaixo, uma observação: para essas primeiras aplicações não é necessário fonte externa, pois o próprio computador fornece energia ao Arduino, antes de ligar no computador verifique todas as ligações novamente;

(2) Com o circuito montado e verificado conecte o Arduino na USB do computador, abra o programa e copie o código descrito na próxima página:



```

int Pinoled = 13;           //led no pino 13
int Botao = 2;              //botao no pino 2
int EstadoBotao = 0;        //Variável para ler o status do pushbutton

void setup()
{
    pinMode(Pinoled, OUTPUT); //Pino do led será saída
    pinMode(Botao, INPUT);   //Pino com botão será entrada
}
void loop()
{
    EstadoBotao = digitalRead(Botao);
    /*
    novo estado do botão vai ser igual ao que
    Arduino ler no pino onde está o botão.
    Poderá ser ALTO (HIGH) se o botão estiver
    Pressionado, ou BAIXO (LOW), se o botão
    estiver solto
    */
    if (EstadoBotao == HIGH) //Se botão estiver pressionado (HIGH)
    {
        digitalWrite(Pinoled, HIGH); // acende o led do pino 13.
    }
    else //se não estiver pressionado
    {
        digitalWrite(Pinoled, LOW); //deixa o led do pino 13 apagado
    }
}

```

(3) Verifique se o código não possui erros, caso tudo esteja devidamente preparado é só pressionar o botão de gravar. É necessário que você siga todos os passos a risca para não correr o risco de danificar a plataforma.

Para pensar...

Através do primeiro experimento pense como seria o programa caso quiséssemos inverter o estado do LED, ou seja, ele só seria ligado caso a chave não estivesse pressionada, e caso contrário o LED apagaria.

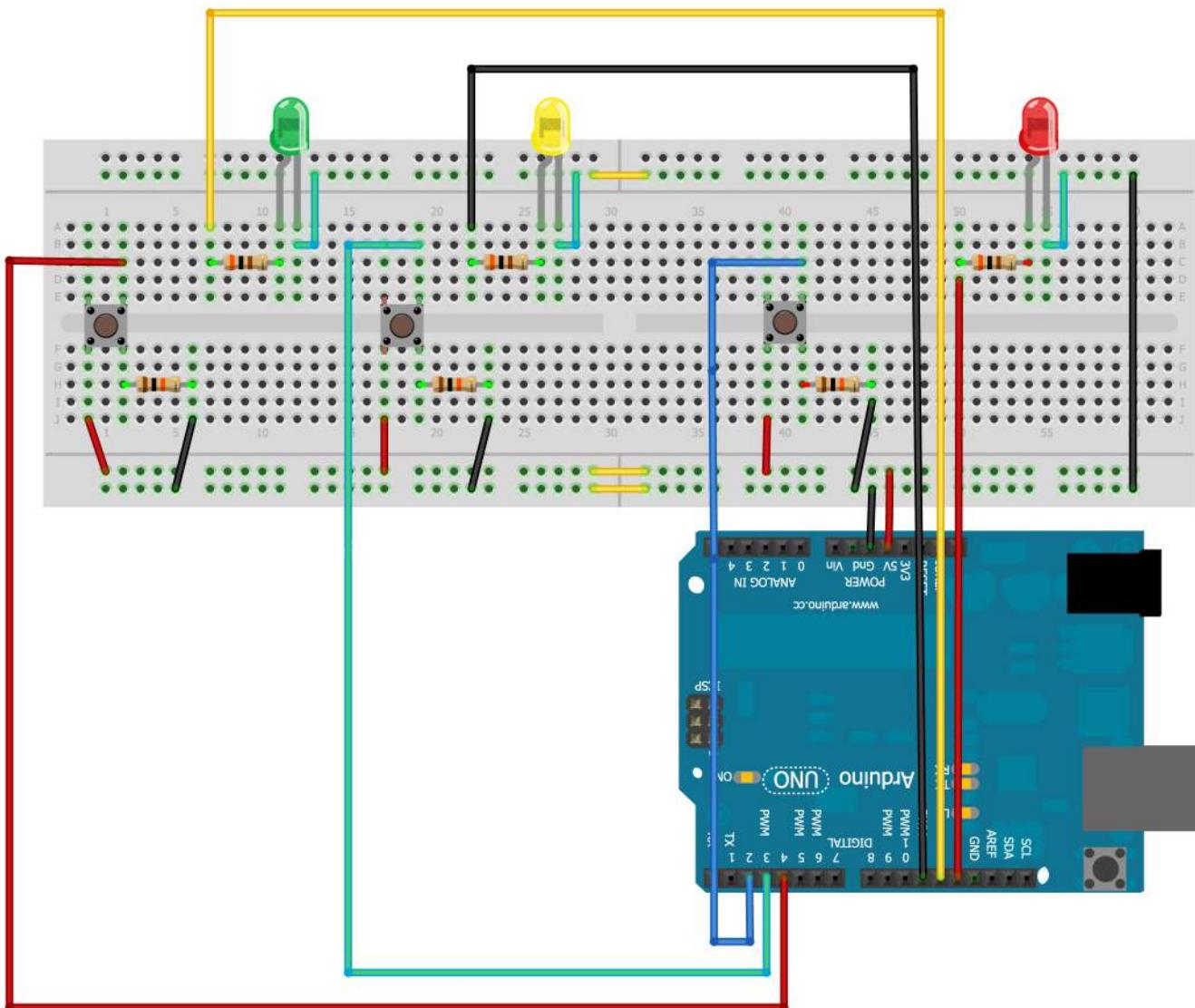
2º Experimento

Objetivo: Controlar 3 LED com 3 botões.

Funcionamento: Iremos seguir o mesmo princípio do experimento anterior, mas agora com 3 LED e 3 push buttons. Cada botão irá acionar um único LED quando for pressionado, e irá desligá-lo caso contrário.

Procedimentos:

(1) Monte o circuito no protoboard da maneira que segue abaixo, note que o circuito será praticamente o mesmo, mas com 2 chaves e 2 LED a mais, antes de ligar o Arduino no computador verifique todas as ligações novamente;



(2) Com o circuito montado e verificado conecte o Arduino na USB do computador, abra o programa e copie o código descrito abaixo:

```

int ledPin1 = 13;           //Led 1 ligado no pino 13
int ledPin2 = 12;           //Led 2 ligado no pino 12
int ledPin3 = 11;           //Led 3 ligado no pino 11
int Botao1 = 2;             //Botão 1 ligado no pino 2
int Botao2 = 3;             //Botão 2 ligado no pino 3
int Botao3 = 4;             //Botão 3 ligado no pino 4
int EstadoBotao1 = 0;       //Botão 1 desligado
int EstadoBotao2 = 0;       //Botão 2 desligado
int EstadoBotao3 = 0;       //Botão 3 desligado

void setup(){                //Configurações dos pinos
    pinMode(ledPin1, OUTPUT); //pino do Led 1 configurado como saída
    pinMode(Botao1, INPUT);  //pino do botão 1 configurado como entrada
    pinMode(ledPin2, OUTPUT); //pino do led 2 configurado como saída
    pinMode(Botao2, INPUT);  //pino do botão 2 configurado como entrada
    pinMode(ledPin3, OUTPUT); //pino do led 3 configurado como saída
    pinMode(Botao3, INPUT);  //pino do botão 3 configurado como entrada
}

void loop(){
    EstadoBotao1 = digitalRead(Botao1); // Lê o estado do botão 1
    EstadoBotao2 = digitalRead(Botao2); // Lê o estado do botão 2
    EstadoBotao3 = digitalRead(Botao3); // Lê o estado do botão 3
    if (EstadoBotao1 == HIGH){ //Se o botão 1 está pressionado
        digitalWrite(ledPin1, HIGH); //acende o led 1
    }
    else{                      //Se não
        digitalWrite(ledPin1, LOW); //apaga o led 1
    }
    if (EstadoBotao2 == HIGH){ //Se o botão 2 está pressionado
        digitalWrite(ledPin2, HIGH); //acende led 2
    }
    else{                      //Se não
        digitalWrite(ledPin2, LOW); //apaga o led 2
    }
    if (EstadoBotao3 == HIGH){ //Se o botão 3 está pressionado
        digitalWrite(ledPin3, HIGH); //acende o led 3
    }
    else{                      //Se não
        digitalWrite(ledPin3, LOW); //apaga led 3
    }
}

```

(3) Verifique se o código não possui erros, caso tudo esteja devidamente preparado é só pressionar o botão de gravar. É necessário que você siga todos os passos a risca para não correr o risco de danificar a plataforma. Note que neste experimento o programa apesar de parecer mais complexo nada mais é do que uma ampliação do primeiro.

Para pensar...

Com base no experimento 2 altere o código para que a primeira chave, da esquerda para a direita, acenda o LED verde, a segunda o LED amarelo, e o LED vermelho só poderá ser aceso com o segundo e o terceiro botão pressionados ao mesmo tempo.

Exercitando o conhecimento

Com o conhecimento adquirido nesse encontro elabore a solução para os seguintes problemas:

1 - Com base no experimento 2, altere o código de programação, para que sejam trocados de lugar os pinos onde estão conectados os LED - troque para os pinos 8, 9, e 10) e as chaves - para os pinos 5, 6, 7.

2 - Você tem 4 botões e 4 LED, sendo que cada botão será responsável por acender uma quantidade específica de LED **simultaneamente**, sendo o 1º botão responsável por acender o primeiro LED, o segundo botão responsável por acender os 2 primeiros LED, o terceiro botão responsável por acender os 3 primeiros LED, e o último que irá acender todos os LED. Implemente o código desse programa.

Quer mais informações!?

Para um maior aprofundamento nos conceitos básicos apresentados, procure por sites sobre a utilização do Arduino, isso irá contribuir muito para a formação de novos conhecimentos com aplicações práticas!

Dúvidas poderão ser encaminhadas aos tutores de vocês!

Listas de Sites interessantes

(1) **Arduino** - site oficial, conceitos e materiais para download.

<<http://playground.arduino.cc//Portugues/HomePage>>

(2) Site com cursos básico e avançado sobre Arduino.

<<http://cesarbastos.wix.com/robotica#!arduino/c13u4>>

(3) **Site do Laboratório de Garagem**, com conceitos diversos sobre a utilização, com projetos e downloads.

<http://www.labdegaragem.com.br/wiki/index.php?title=Sobre_Arduino>



Oficinas Tecnológicas para alunos do Ensino Médio

Prof. Dr. Leonardo Mesquita

mesquita@feg.unesp.br

Coordenador do Projeto

Prof. Dr. Galeno José de Sena

gsena@feg.unesp.br

Prof. Dr. Marco Aurélio Alvarenga Monteiro

marco.aurelio@feg.unesp.br

Prof. José Marcelo de A. Wendling Jr.

jmarcelo@feg.unesp.br

Felipe Barbosa da Silva

Ismael de Almeida Júnior

Letícia Miranda de França Mota

Renan Moura Santana

Samuel José de Carvalho