

CURSO DE ROBÓTICA 2

NOVOS CONCEITOS DE PROGRAMAÇÃO



PET-CoCE



REVISÃO

ELETRICIDADE



REVISÃO

ELETRICIDADE

→ Potenciômetro



Fonte:
<http://www.baudaeletronica.com.br/potenciometro-linear-d-e-10k-10000.html>

→ LEDs



Fonte: <https://oscarliang.com/use-led-in-robot-projects/>

→ Push button



Fonte:
<https://www.boxeelectronica.com/pt/botoes-switch-on-off/18-botao-de-pressao-6x6x5mm.html>

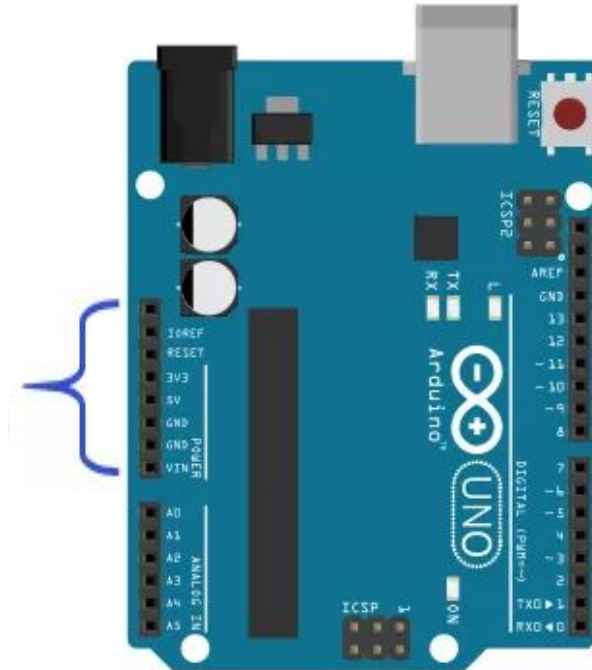
REVISÃO

PINOS DE ALIMENTAÇÃO?

5V: Fornece uma tensão de 5V em relação à referência.

GND: Do inglês, GROUND (terra). Referência: 0V.

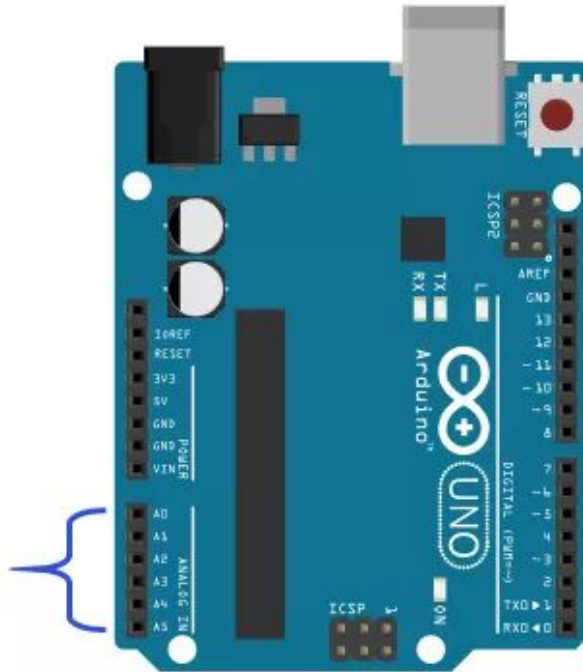
Vin: Pino usado para alimentar a placa arduino.



REVISÃO

ENTRADAS ANALÓGICAS E SAÍDAS PWM?

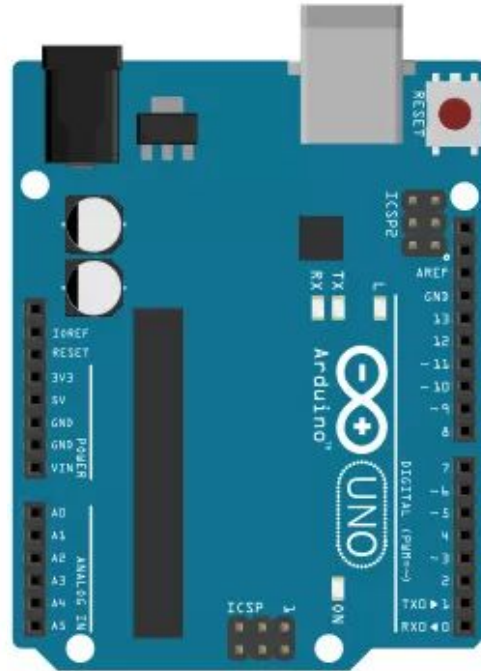
Entradas analógicas: são usadas para lerem valores analógicos, como potenciômetros, sensores de temperatura etc.



Saídas PWM: simula o comportamento de um sinal analógico. Utilizado para variar o brilho do LED por exemplo. Possuem um “tio” (~) na frente.

REVISÃO: ARDUINO

ENTRADAS E SAÍDAS DIGITAIS



Entradas e saídas digitais: trabalham apenas com dois níveis de tensão, 0 e 5V, comumente chamados de 0 e 1. Pode-se usar como analogia um interruptor.

REVISÃO: ARDUINO

MINI EXERCÍCIO

```
void setup() {           // tudo que está dentro das chaves será executado apenas uma vez logo que o arduino é ligado
  ...
}
void loop() {            // tudo que está dentro das chaves {} será executado enquanto placa estiver alimentada
  ...
  ...
  ...
  ...
}
```

REVISÃO: ARDUINO

MINI EXERCÍCIO

```
const int LED = 0; // define o pino no qual o led está conectado
void setup() {      // tudo que está dentro das chaves será executado apenas uma vez logo que o arduino é ligado
...
}
void loop() {       // tudo que está dentro das chaves {} será executado enquanto placa estiver alimentada
...
...
...
...
}
```


REVISÃO: ARDUINO

MINI EXERCÍCIO

```
const int LED = 0; // define o pino no qual o led está conectado
void setup() {      // tudo que está dentro das chaves será executado apenas uma vez logo que o arduino é ligado
    pinMode(LED, OUTPUT); // define o pino_do_LED como uma saída
}
void loop() {       // tudo que está dentro das chaves {} será executado enquanto placa estiver alimentada
    ...
    ...
    ...
    ...
}
```

REVISÃO: ARDUINO

MINI EXERCÍCIO

```
const int LED = 0; // define o pino no qual o led está conectado
void setup() {      // tudo que está dentro das chaves será executado apenas uma vez logo que o arduino é ligado
    pinMode(LED, OUTPUT); // define o pino_do_LED como uma saída
}
void loop() {       // tudo que está dentro das chaves {} será executado enquanto placa estiver alimentada
    digitalWrite(LED, HIGH); // “escreve” nível alto no pino do LED, isto é, liga o LED
    ...
    ...
    ...
}
```

REVISÃO: ARDUINO

MINI EXERCÍCIO

```
const int LED = 0; // define o pino no qual o led está conectado
void setup() {      // tudo que está dentro das chaves será executado apenas uma vez logo que o arduino é ligado
    pinMode(LED, OUTPUT); // define o pino_do_LED como uma saída
}
void loop() {       // tudo que está dentro das chaves {} será executado enquanto placa estiver alimentada
    digitalWrite(LED, HIGH); // “escreve” nível alto no pino do LED, isto é, liga o LED
    delay(500);          // pausa no estado em que se encontra durante 500ms, ou seja, 0,5s
    ...
    ...
}
```

REVISÃO: ARDUINO

MINI EXERCÍCIO

```
const int LED = 0; // define o pino no qual o led está conectado
void setup() {      // tudo que está dentro das chaves será executado apenas uma vez logo que o arduino é ligado
    pinMode(LED, OUTPUT); // define o pino_do_LED como uma saída
}
void loop() {       // tudo que está dentro das chaves {} será executado enquanto placa estiver alimentada
    digitalWrite(LED, HIGH); // “escreve” nível alto no pino do LED, isto é, liga o LED
    delay(500);              // pausa no estado em que se encontra durante 500ms, ou seja, 0,5s
    digitalWrite(LED, LOW);  // “escreve” nível baixo no pino do LED, isto é, desliga o LED
    delay(500);              // pausa no estado em que se encontra durante 500ms, ou seja, 0,5s
}
```

NOVAS COISAS LEGAIS EM PROGRAMAÇÃO

E ÚTEIS!!!

(MAS ANTES UMA REVISÃOZINHA)

COISAS LEGAIS EM PROGRAMAÇÃO

REVISÃO

→ Variáveis

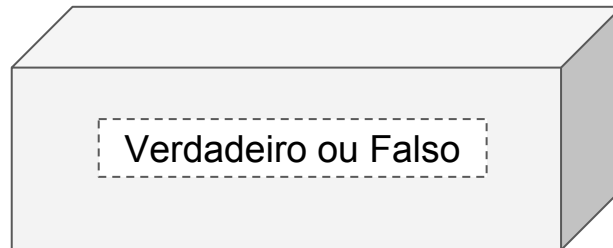
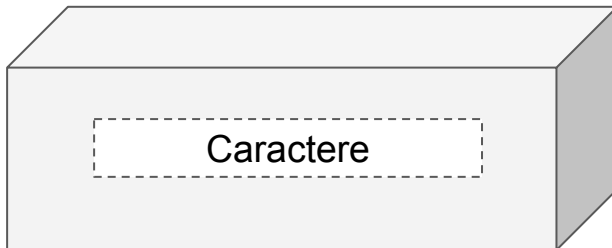
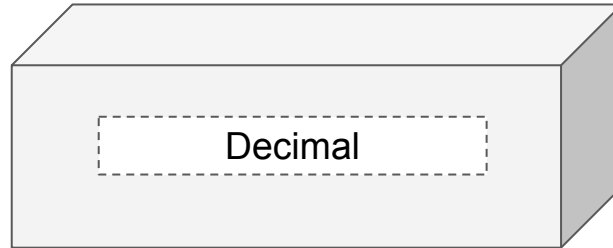

Caixa
tem um tipo



COISAS LEGAIS EM PROGRAMAÇÃO

REVISÃO

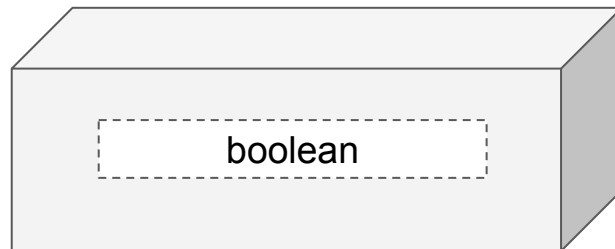
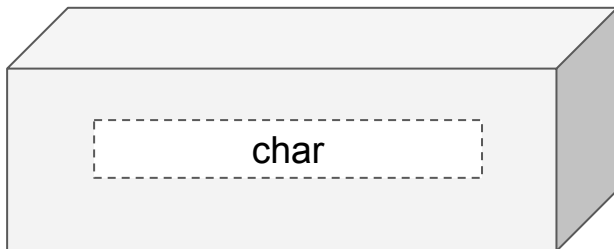
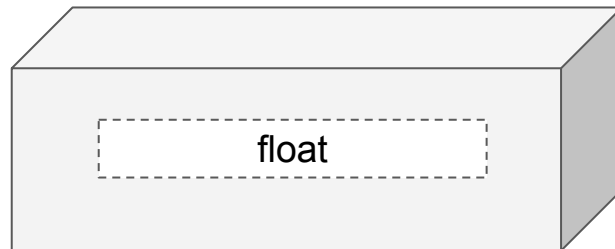
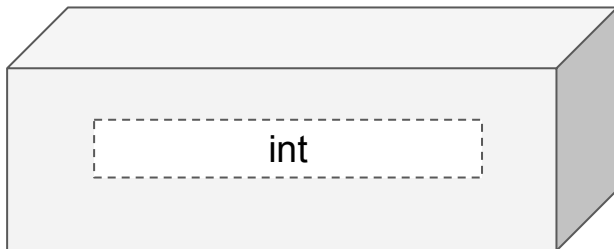
→ Variáveis



COISAS LEGAIS EM PROGRAMAÇÃO

REVISÃO

→ Variáveis



COISAS LEGAIS EM PROGRAMAÇÃO

REVISÃO

→ Variáveis

```
int idade;
float altura;
char primeira_letra_do_meu_nome;
boolean ligado;

void setup() {
  Serial.begin(9600);  //inicializa a serial

  idade = 20;
  altura = 1.58;
  primeira_letra_do_meu_nome = 'g';
  ligado = true;
```

```
Serial.println(idade);
Serial.println(altura);
Serial.println(primeira_letra_do_meu_nome);
Serial.println(ligado);
```

```
idade = 21.5;           //21
altura = 12;            //12.00
primeira_letra_do_meu_nome = 'g';  //G
ligado = 20;            //TRUE
```

```
Serial.println("Teste de novo.");
Serial.println(idade);
Serial.println(altura);
Serial.println(primeira_letra_do_meu_nome);
Serial.println(ligado);
```

```
}
```

COISAS LEGAIS EM PROGRAMAÇÃO

REVISÃO

→ Funções

```
void setup() {}  
void loop() {}
```

Exemplo: uma função que você dá dois números e ela te retorna a multiplicação deles.

```
int multiplica(int a, int b){  
    int resultado;  
    resultado = a*b;  
    return resultado;  
}
```

COISAS LEGAIS EM PROGRAMAÇÃO

IF E ELSE

→ Se e senão

```
if(TESTE){  
    FAÇA ISSO;  
}  
else{  
    FAÇA ISSO;  
}
```

Exemplo: uma função que recebe dois números e retorna o maior deles.

```
int maior(int a, int b){  
    int resposta;  
    if(a>b){  
        resposta = a;  
    }  
    else{  
        resposta = b;  
    }  
    return resposta;  
}
```

COISAS LEGAIS EM PROGRAMAÇÃO

LOOPS

→ While

```
while(TESTE){  
    FAÇA ISSO;  
}
```

Exemplo: uma função que imprime na tela todos os números entre 0 e 10.

```
void contaWhile(){  
    int i = 0;  
    while(i <= 10){  
        Serial.println(i);  
        i++;  
    }  
}
```

COISAS LEGAIS EM PROGRAMAÇÃO

LOOPS

→ For

```
for(INICIALIZAÇÃO; CONDIÇÃO; INCREMENTO){  
    FAÇA ISSO;  
}
```

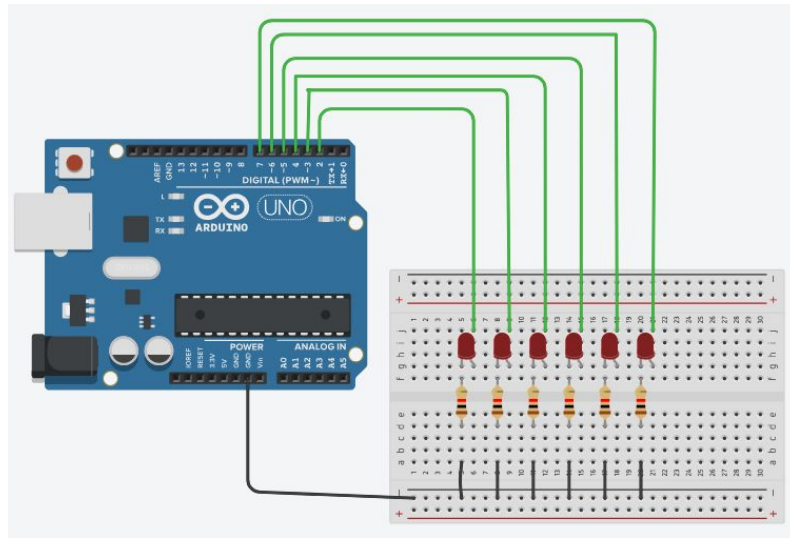
Exemplo: uma função que imprime na tela todos os números entre 0 e 10.

```
void conta(){  
    for(int i=0; i<=10; i++){  
        Serial.println(i);  
    }  
}
```

COISAS LEGAIS EM PROGRAMAÇÃO

EXERCÍCIO

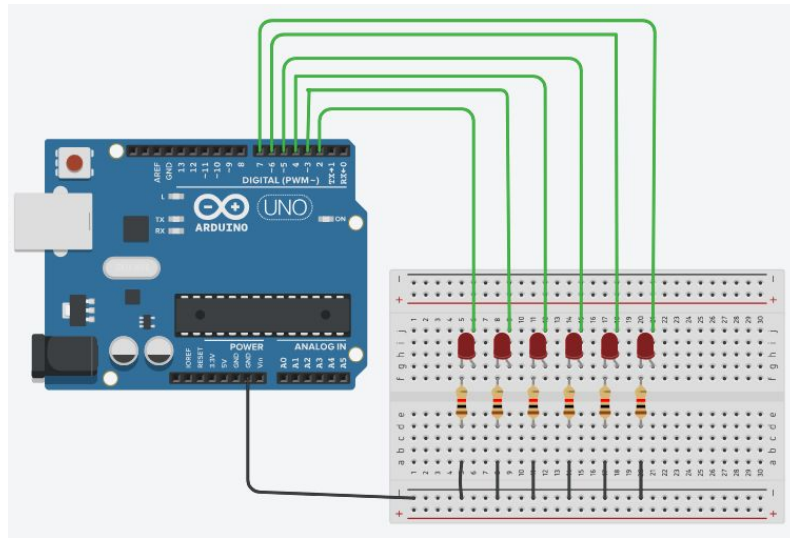
- Montar o seguinte circuito e fazer os LEDs acenderem e apagarem em ordem (quando o LED 1 apaga, o LED 2 acende, em seguida o LED 2 apaga e o LED 3 acende):



COISAS LEGAIS EM PROGRAMAÇÃO

EXERCÍCIO

- A partir do mesmo circuito fazer os LEDs acenderem e apagarem na ordem inversa (quando o LED 3 apaga, o LED 2 acende, em seguida o LED 2 apaga e o LED 1 acende):



COISAS LEGAIS EM PROGRAMAÇÃO

RESPOSTA

```
int timer = 100; // Define o tempo de delay entre um
                // LED e outro

void setup() {
    // Loop usado para inicializar os pinos com saídas:
    for (int thisPin = 2; thisPin < 8; thisPin++) {
        pinMode(thisPin, OUTPUT);
    }
}

void loop() {
    // loop do primeiro para o último LED:
    for (int thisPin = 2; thisPin < 8; thisPin++) {
        // acende o LED:
        digitalWrite(thisPin, HIGH);
        delay(timer);
        // apaga o LED:
        digitalWrite(thisPin, LOW);
    }
}
```

```
// Loop do último para o primeiro LED:
    for (int thisPin = 7; thisPin >= 2;
        thisPin--) {
        //acende o pino:
        digitalWrite(thisPin, HIGH);
        delay(timer);
        // apaga o pino:
        digitalWrite(thisPin, LOW);
    }
}
```




**ESTE TRABALHO ESTÁ LICENCIADO SOB UMA LICENÇA CREATIVE
COMMONS ATRIBUIÇÃO 4.0 INTERNACIONAL.
PARA VER UMA CÓPIA DESTA LICENÇA, VISITE
[HTTP://CREATIVECOMMONS.ORG/LICENSES/BY/4.0/](http://creativecommons.org/licenses/by/4.0/)**