

A9

Arduino

Sensor e LED

Quem quer fazer seu próprio código?



**O desafio agora é fazer fade
em um LED a partir de um
sensor de luz!**

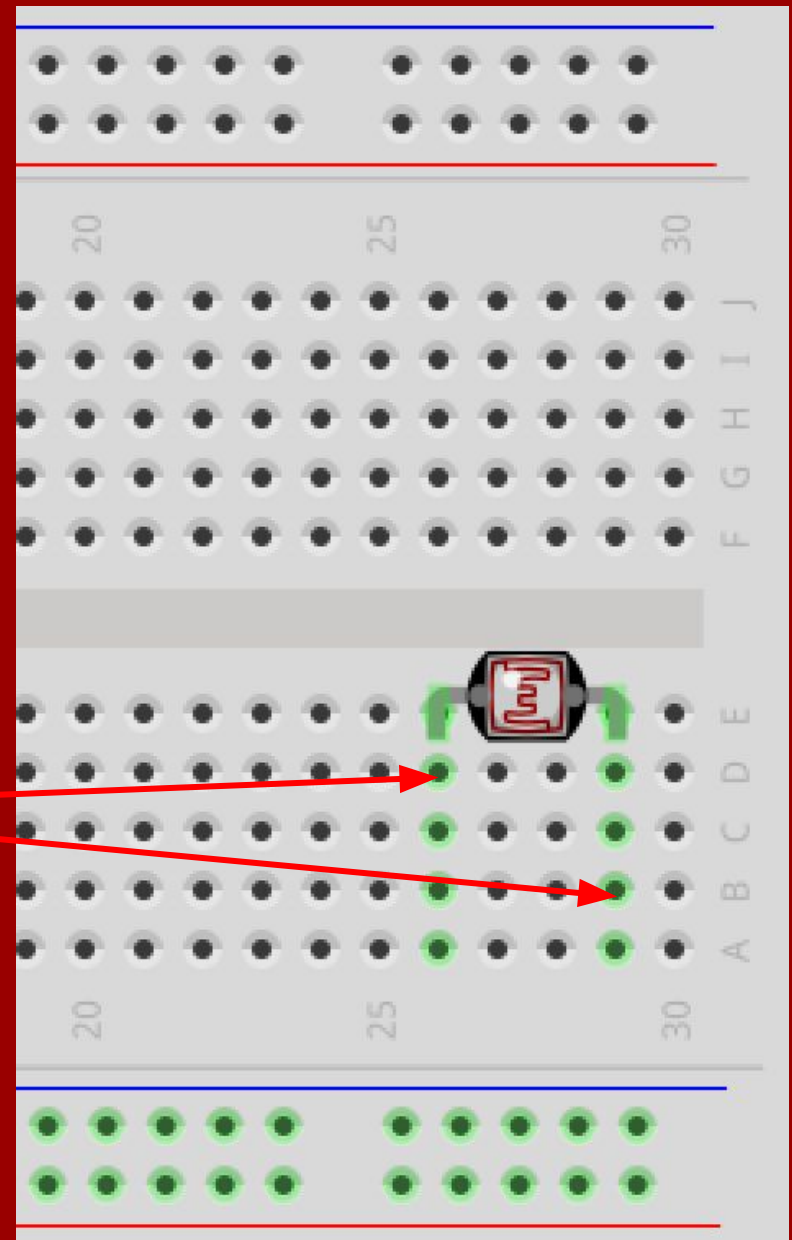
Sensor de luz???



Pois é... o nome oficial é Fotoresistor. É parecido com o outro resistor que usamos, mas a energia que ele deixa passar muda de acordo com a quantidade de luz em cima dele. Quanto mais luz, mais energia passa.

E como a gente
liga ele no
Arduino?

Primeiro, escolha
duas colunas
vazias na
protoboard.

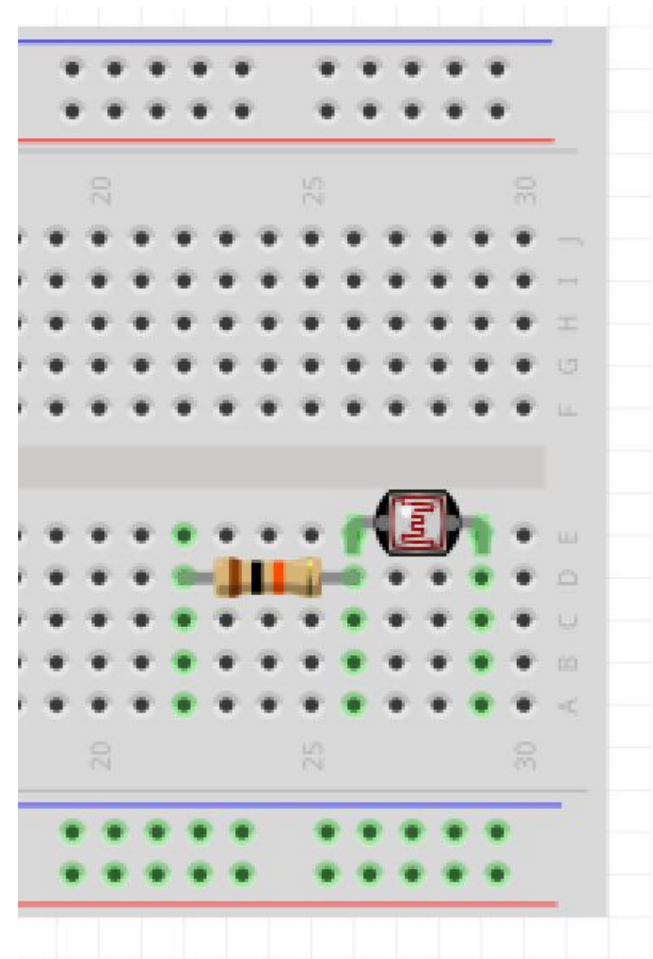


Agora colocamos um resistor de 10 kohm em série com o fotoresistor.



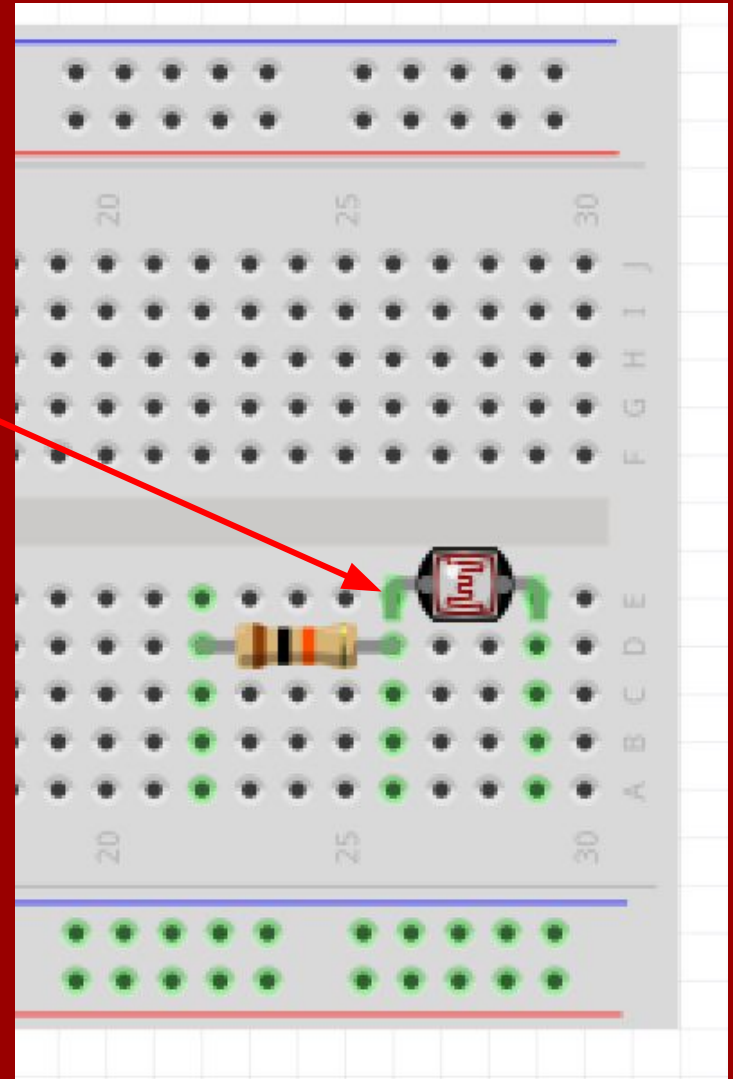
10 kohm: marrom, preto, laranja

Em "série"?
Como assim?

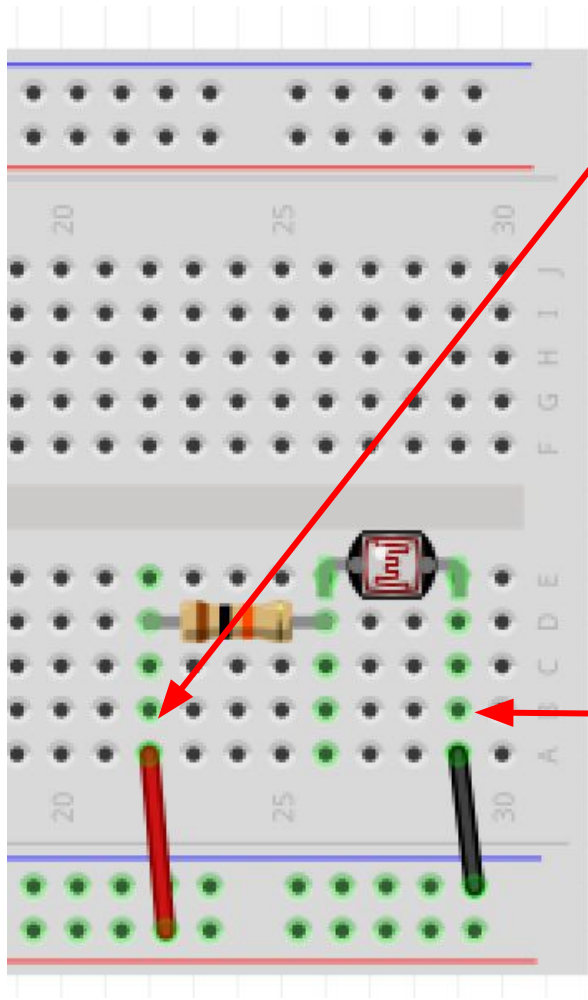


Quando dois componentes compartilham somente **um ponto** de contato, eles estão "em série".

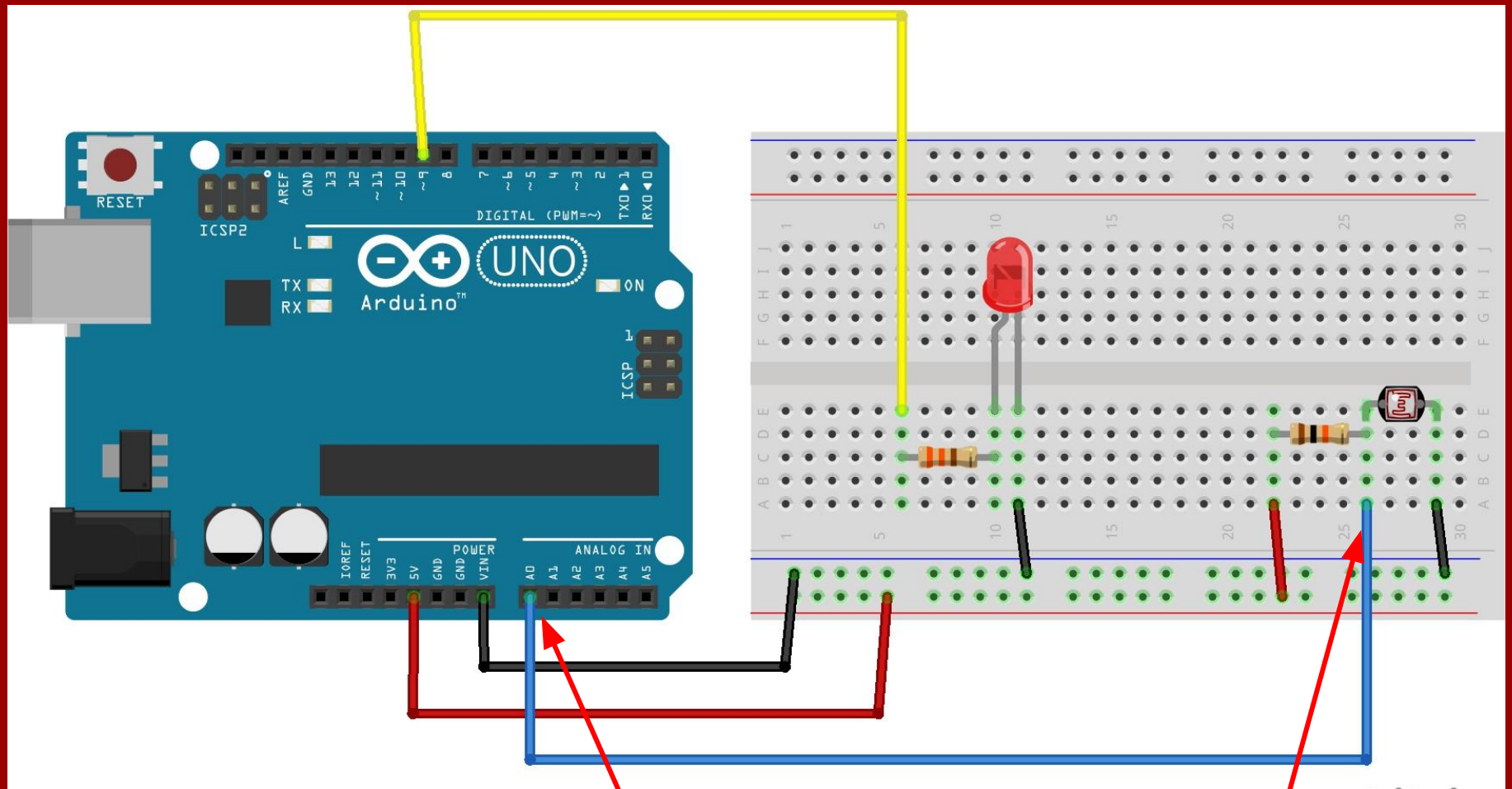
A corrente vai passar por um, e depois pelo outro.



Agora conectamos a pata livre do resistor ao +5V...



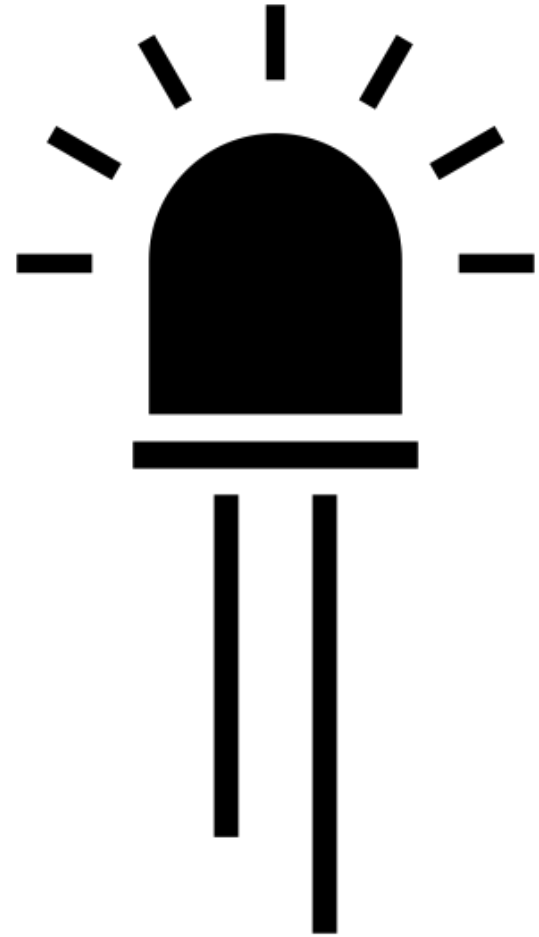
... e a pata livre do fotoresistor ao GND.



Por fim, conectamos um jumper entre a **coluna** dos dois componentes e a porta **A0**.

Agora vamos integrar
nosso LED ao circuito
do sensor.

O objetivo é acender
um LED conforme a
quantidade de luz no
nosso sensor. Quanto
mais luz no sensor,
mais fraco deve ficar o
LED.





Vamos ao código!

Por onde começar um código?

Pelas variáveis, setup ou loop?



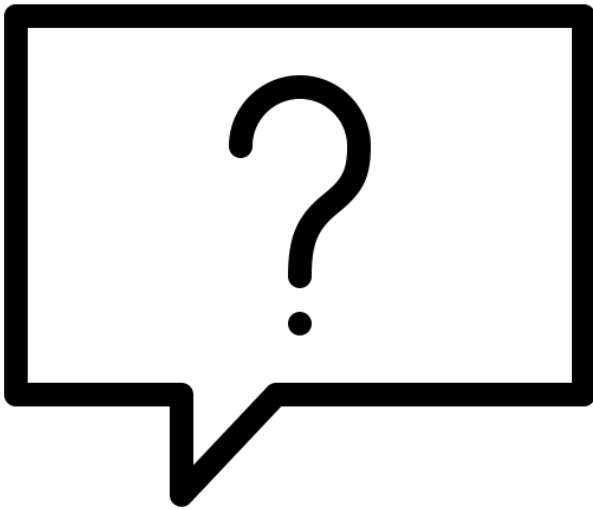
Resposta: pela LÓGICA!

Para fazer o fade no LED a partir do fotoresistor, precisamos:

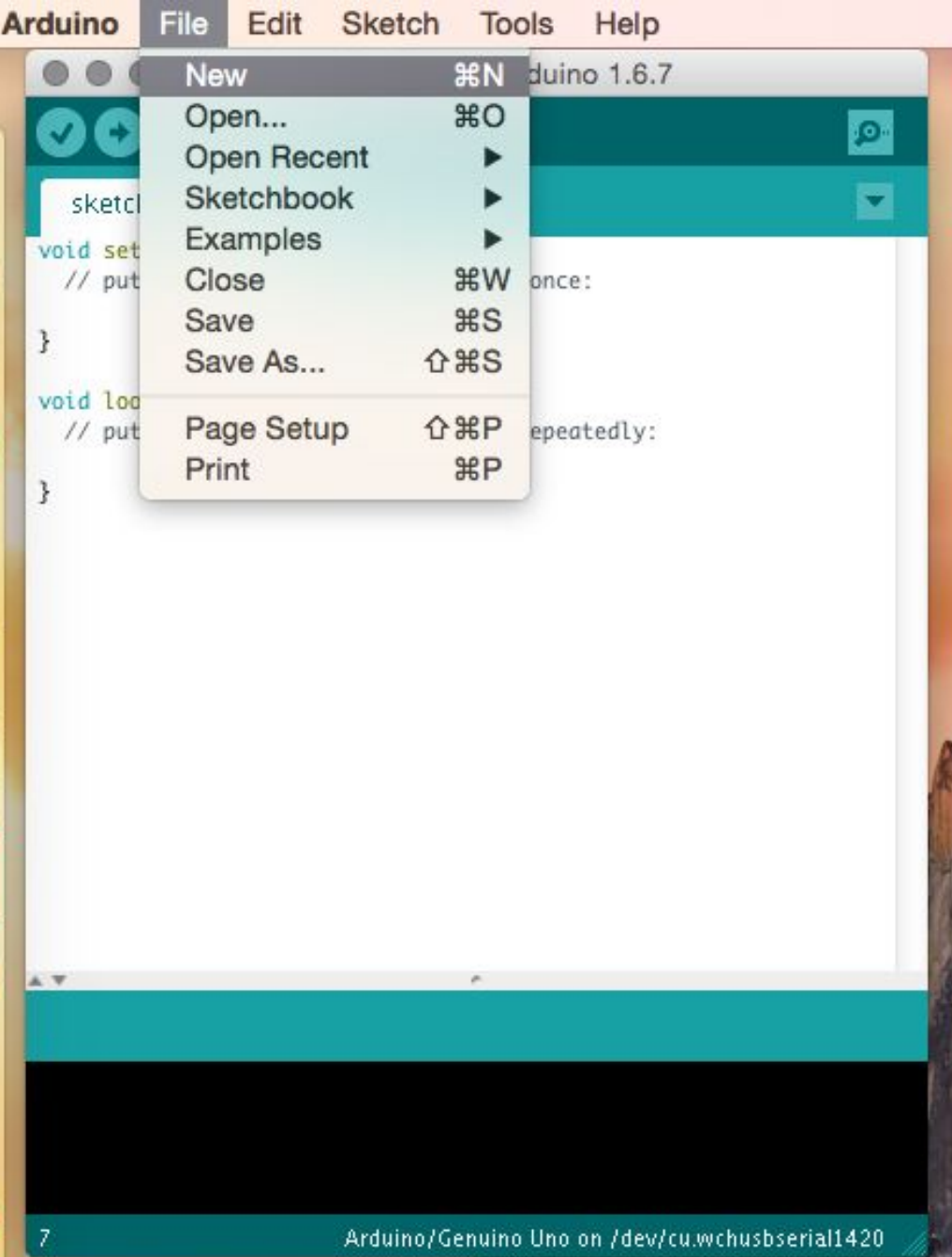
1. fazer a leitura do valor do fotoresistor;

2. usar esse valor para determinar o brilho do LED (a famosa variável **brilho**).

Pergunto:



Estas instruções
fazem parte do
código do setup
ou do loop???



**Vamos criar
um novo
documento**

Arquivo>Novo

Nossa primeira linha deste código seria...

O fotoresistor oferece valores digitais ou analógicos?
Logo, qual função utilizar?

Que parâmetro esta função espera receber?

O que precisamos armazenar?

Nossa primeira linha deste código seria...

O fotoresistor oferece valores digitais ou analógicos?
Logo, qual função utilizar?

Que parâmetro esta função espera receber?

O que precisamos armazenar?

```
void loop() {  
  // put your main code here, to run repeatedly:  
  int valorDoSensor = analogRead(A0);  
}
```

**E a segunda linha do código?
O que precisamos fazer com esse valor
do sensor?**

Lembra que para fazer um fade no LED se utiliza a função **analogWrite()**, e ela espera DOIS parâmetros? Um diz respeito a porta e o outro ao valor do brilho do LED.

E a segunda linha do código? Faz sentido?

```
void loop() {  
  // put your main code here, to run repeatedly:  
  int valorDoSensor = analogRead(A0);  
  analogWrite(9, valorDoSensor);  
}
```

Quanto mais luz cai no sensor, mais baixa a sua resistência, e menor o valor lido pela função **analogRead()**.

Hum... ainda falta alguma coisa, né?

Temos de passar as configurações iniciais para o sistema na função **setup()**.

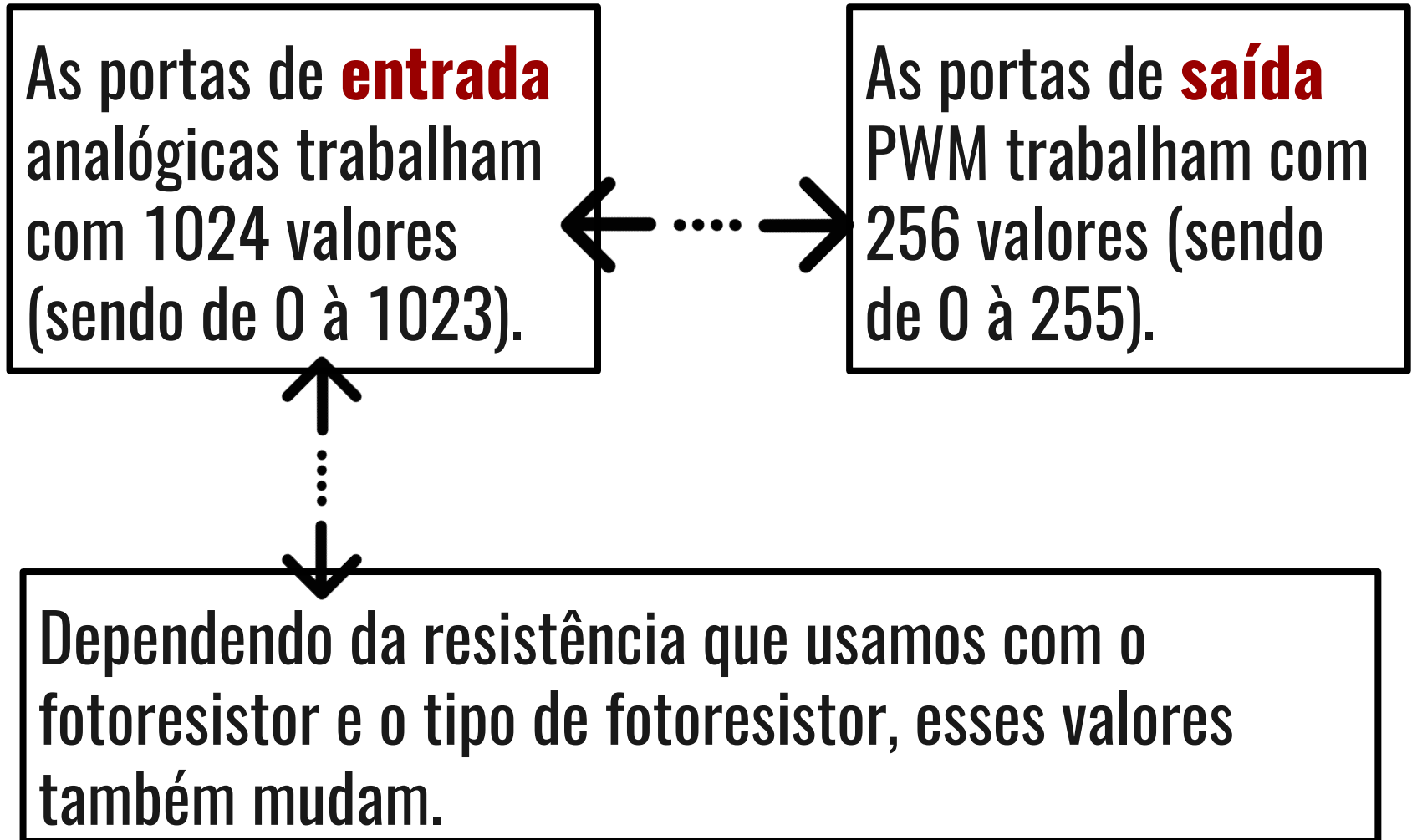
```
void setup() {  
    // put your setup code here, to run once:  
    pinMode(9, OUTPUT);  
}
```

**E agora, o código está
funcionando com o circuito?**

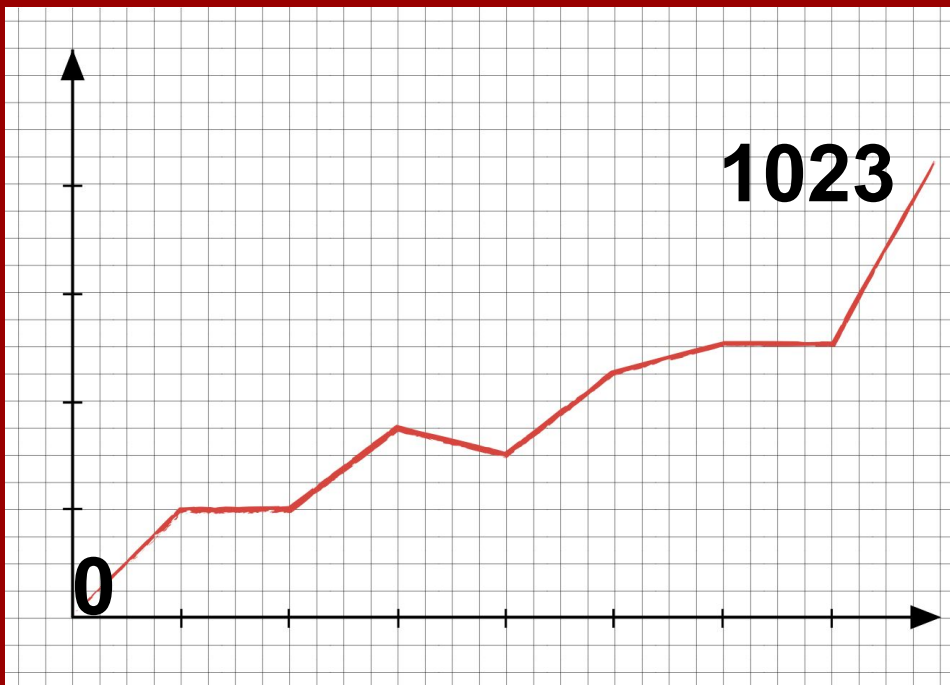


Funciona, mas
tá meia boca,
né? O que será
que falta fazer
para funcionar?

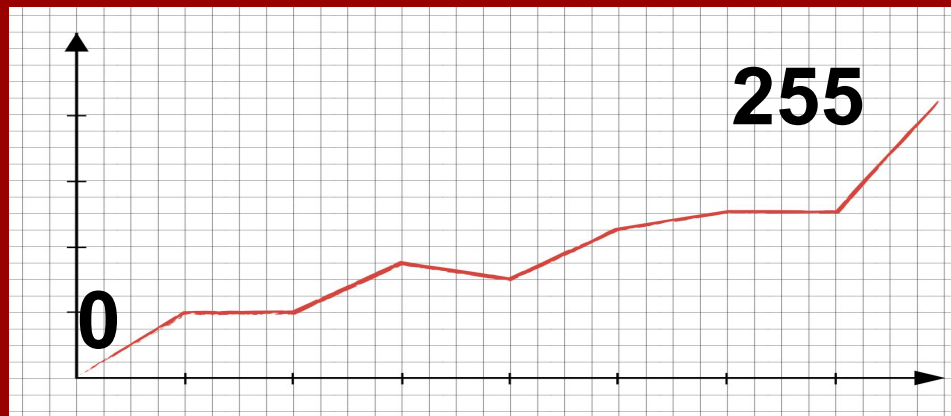
Tudo é uma questão de ajustar escala...



Como re-escalar? Matemática e chute!

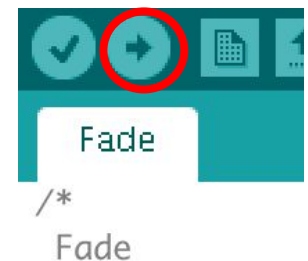


Queremos manter a mesma forma do sinal, mas ter valores menores.



```
void loop() {  
  // put your main code here, to run repeatedly:  
  int valorDoSensor = analogRead(A0);  
  analogWrite(9, valorDoSensor / 4);  
}
```

1024 é um valor 4 vezes maior que 256. Podemos começar dividindo o valor que entra pela porta **A0** por 4 antes de passar para a função **analogWrite()**.



Como re-escalar? Matemática e chute!

```
void loop() {  
  // put your main code here, to run repeatedly:  
  int valorDoSensor = analogRead(A0);  
  analogWrite(9, valorDoSensor / 6);  
}
```

Se não deu certo, tente dividir por um número maior (6 ou 8), para ver se o sensor fica mais sensível.

**Pronto, você já sabe o básico sobre o
Arduino e suas funções de leitura e
controle de sinais analógicos e
digitais.**

