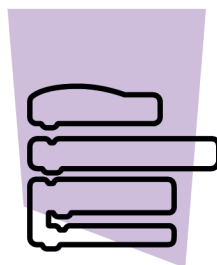
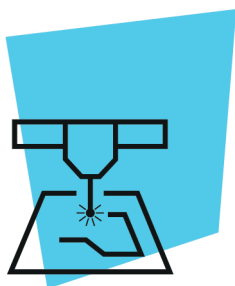
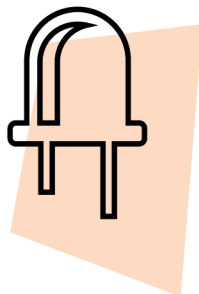
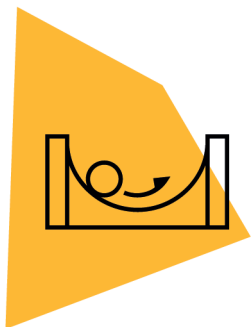
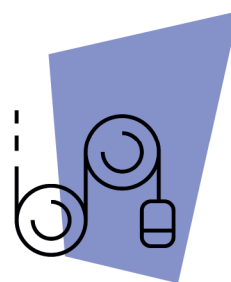


# ROBÓTICA

Módulo 1



## Sensor de Temperatura

AULA 29

**GOVERNADOR DO ESTADO DO PARANÁ**

Carlos Massa Ratinho Júnior

**SECRETÁRIO DE ESTADO DA EDUCAÇÃO**

Renato Feder

**DIRETOR DE TECNOLOGIA E INOVAÇÃO**

Andre Gustavo Souza Garbosa

**COORDENADOR DE TECNOLOGIAS EDUCACIONAIS**

Marcelo Gasparin

**Produção de Conteúdo**

Andrea da Silva Castagini Padilha

Cleiton Rosa

Simone Sinara de Souza

**Revisão Textual**

Adilson Carlos Batista

**Projeto Gráfico e Diagramação**

Edna do Rocio Becker

2021



Este trabalho está licenciado com uma Licença Creative Commons  
Atribuição NãoComercial - Compartilha Igual 4.0 Internacional

Aula 01	Por Que Robótica?
Aula 02	Tensão, Corrente e Resistência
Aula 03	Kit de Robótica
Aula 04	Arduino Uno R3
Aula 05	Softwares Arduino IDE e mBlock
Aula 06	Portas Digitais
Aula 07	Circuito Elétrico
Aula 08	LED e Resistor
Aula 09	Semáforo [Carros]
Aula 10	Semáforo [Cruzamento Carros]
Aula 11	Semáforo [Pedestres]
Aula 12	Semáforo [Cruzamento Carros + Pedestres]
Aula 13	Push Button
Aula 14	Feedbacks + Inventário I
Aula 15	Semáforo [Carros + Pedestres com Botão]
Aula 16	Display 7 Segmentos
Aula 17	Fonte DC + Plug P4
Aula 18	Portas PWM
Aula 19	LED Fade-In
Aula 20	LED Fade-Out
Aula 21	Super Máquina 80's
Aula 22	Super Máquina 2008
Aula 23	Potenciômetro
Aula 24	Buzzer Passivo
Aula 25	LED RGB
Aula 26	Arco-Iris
Aula 27	Sensor LDR
Aula 28	Feedbacks + Inventário II
Aula 29	Sensor de Temperatura
Aula 30	Sensor de Obstáculo IR
Aula 31	Controle Motor DC
Aula 32	Kit Chassi 2WD Robô
Aula 33	Seguidor de Linha
Aula 34	Sensor de Distância
Aula 35	Sensor de Estacionamento
Aula 36	Display LCD 16x2
Aula 37	Trena Digital
Aula 38	Robô Sumô [Estrutura]
Aula 39	Robô Sumô [Programação + Treinamento I]
Aula 40	Robô Sumô [Programação + Treinamento II]
Aula 41	Disputa de Sumôs
Aula 42	Feedbacks + Inventário III

Aula 28  
Feedbacks  
+ Inventário II

Aula 29  
Sensor de  
Temperatura

Aula 30  
Sensor de  
Obstáculo IR

# Sumário

<b>Introdução</b>	<b>2</b>
<b>Objetivos desta Aula</b>	<b>2</b>
<b>Competências Gerais Previstas na BNCC</b>	<b>3</b>
<b>Habilidades do Século XXI a Serem Desenvolvidas</b>	<b>4</b>
<b>Lista de Materiais</b>	<b>4</b>
<b>Roteiro da Aula</b>	<b>5</b>
1. Contextualização	5
2. Montagem e Programação	6
3. Feedback e Finalização	12
<b>Videotutorial</b>	<b>13</b>



## Introdução

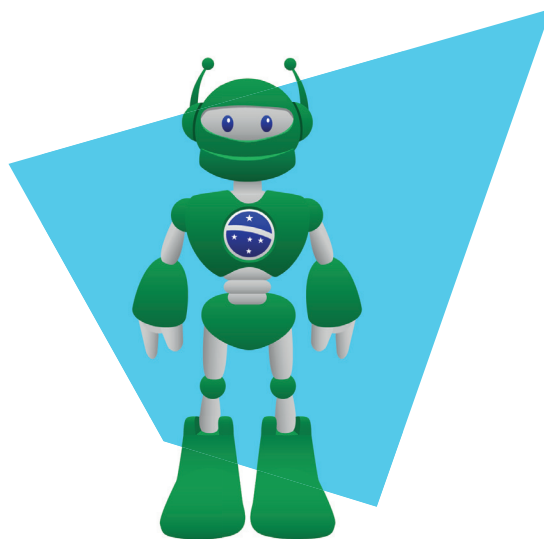
A temperatura é uma grandeza física escalar que representa o grau de agitação das moléculas presente em um ambiente ou corpo, portanto a temperatura caracteriza um estado térmico. Para realizarmos aferições da temperatura, utilizamos um instrumento chamado termômetro. Em geral, os termômetros possuem escalas termométricas relacionadas aos pontos de fusão e ebulição de uma substância, por exemplo, as escalas Celsius e Fahrenheit, mas há também a escala Kelvin, denominada absoluta, pois não depende de medida nem da substância ou propriedade utilizada para medi-la.

Nesta aula, utilizaremos o sensor de temperatura LM35 para medir a temperatura do ambiente.



## Objetivos desta Aula

- Conhecer o sensor de temperatura LM35;
- Programar o sensor LM35 para indicar a temperatura ambiente.





## Competências Gerais Previstas na BNCC

**[CG02]** - Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.

**[CG04]** - Utilizar diferentes linguagens – verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e digital –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.

**[CG05]** - 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.

**[CG09]** - Exercitar a empatia, o diálogo, a resolução de conflitos e a cooperação, fazendo-se respeitar e promovendo o respeito ao outro e aos direitos humanos, com acolhimento e valorização da diversidade de indivíduos e de grupos sociais, seus saberes, identidades, culturas e potencialidades, sem preconceitos de qualquer natureza.

**[CG10]** - Agir pessoal e coletivamente com autonomia, responsabilidade, flexibilidade, resiliência e determinação, tomando decisões com base em princípios éticos, democráticos, inclusivos, sustentáveis e solidários.



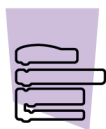
## Habilidades do Século XXI a Serem Desenvolvidas

- Pensamento crítico;
- Afinidade digital;
- Resiliência;
- Resolução de problemas;
- Colaboração;
- Comunicação.



## Lista de Materiais

- 01 Placa Protoboard;
- 01 Placa de Arduino Uno R3;
- 01 Cabo USB;
- 03 Jumpers Macho-Macho;
- 01 Sensor de temperatura LM35;
- Software mBlock ou Arduino IDE;
- 01 Notebook.



## Roteiro da Aula

### 1. Contextualização (15min):

Você já deve ter percebido que, conforme a temperatura local, seu corpo responde através de sensações de frio, arrepiando os pelos, ou calor, com produção de suor. Estas sensações são geradas através do sensor tato, encontrado na pele, e repleto de terminações nervosas capazes de captar estímulos térmicos, mecânicos, entre outros. Além do sensor tato, que outros sensores você identifica no corpo humano?

Isso mesmo! Podemos dizer que os cinco órgãos do sentido (tato, olfato, audição, paladar e visão) são sensores presentes em nosso corpo capazes de captar informações do ambiente, direcionando-as ao cérebro para que esse tome a decisão adequada e os oriente a reagirem de acordo com cada situação.

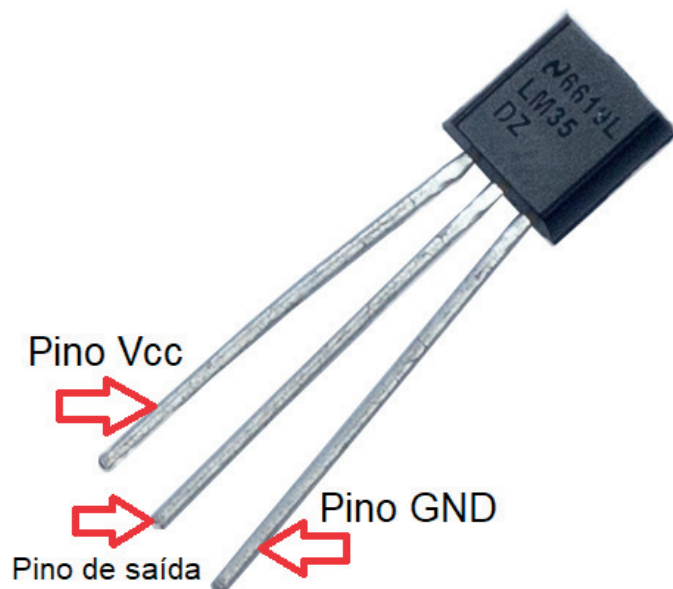
Agora, vamos pensar nos equipamentos eletroeletrônicos. Você saberia dizer como é mensurada a energia térmica presente nesses equipamentos?

Entre os diversos componentes eletrônicos utilizados para a leitura da temperatura de um circuito, podemos citar o **sensor de temperatura LM35**.

Este sensor é um transistor (figura 1) que funciona como leitor de temperatura em uma faixa de medição variável de  $-55^{\circ}\text{C}$  a  $150^{\circ}\text{C}$  e com tensão de saída linear proporcional à temperatura em que se encontra no momento, tendo o sinal de  $10\text{ mV}/^{\circ}\text{C}$  (dez milivolt por grau Celsius). Isto significa que cada  $10\text{ mV}$  de tensão na saída, representa  $1^{\circ}\text{C}$  de temperatura.



Figura 1 - Sensor de temperatura LM35

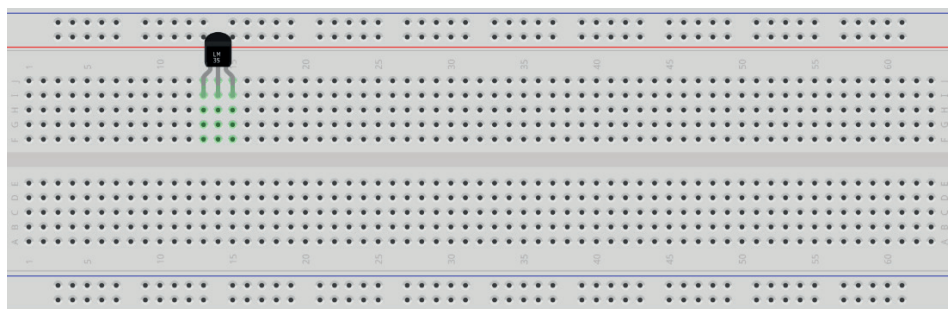


Podemos encontrar esse modelo de sensor de temperatura em fontes de alimentação, sistemas de ar-condicionado, geladeiras, gerenciamento de baterias, entre outros produtos.

## 2. Montagem e Programação (60min):

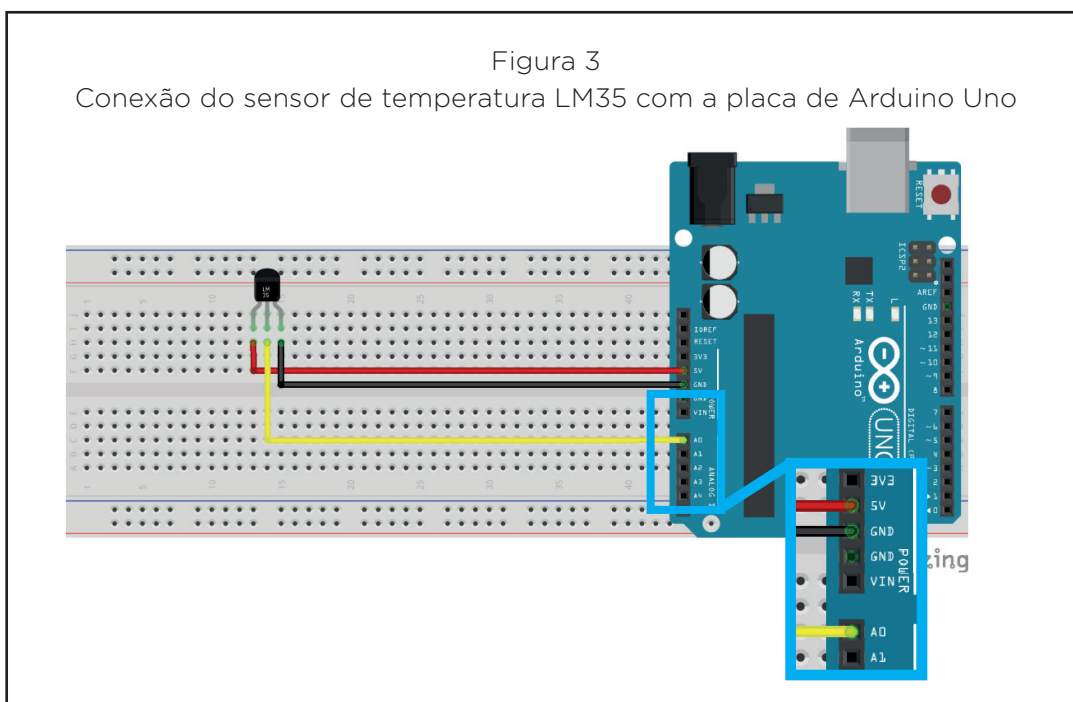
Vamos iniciar a montagem dos componentes eletrônicos. Insira na placa Protoboard o sensor de temperatura LM35 com a face plana voltada para você, conforme mostra a figura 2.

Figura 2 - Encaixe do sensor de temperatura LM35 na Protoboard



fritzing

Conecte 3 jumpers Macho-Macho entre os terminais do sensor de temperatura e a placa de Arduino, respeitando a seguinte sequência: 1 jumper ligado ao terminal esquerdo do sensor de temperatura e a porta 5V do Arduino; 1 jumper entre o terminal central do sensor e a porta analógica A0 da placa de Arduino; e 1 jumper entre o terminal direito do sensor e o pino GND do Arduino, como mostra a figura 3.



Agora, vamos programar!

Com os componentes eletrônicos montados, vamos programar, por codificação e por blocos, o nosso protótipo.

#### i. Linguagem de programação por código

Para iniciar a programação, conecte a placa Arduino ao computador, através de um cabo USB, para que ocorra a comunicação entre a placa microcontroladora e o software Arduino IDE.

No software IDE, escreva ou copie e cole o código-fonte de programação, conforme apresentado no quadro 1.

**Quadro 1 - Código-fonte da programação na linguagem do Arduino (Wiring)**

```
/* Sensor de temperatura usando o LM35 */

const int LM35 = A0; /* Define o pino de dados do sensor LM35
*/
float Temperatura = 0; /* Variável que armazenará a
temperatura medida */

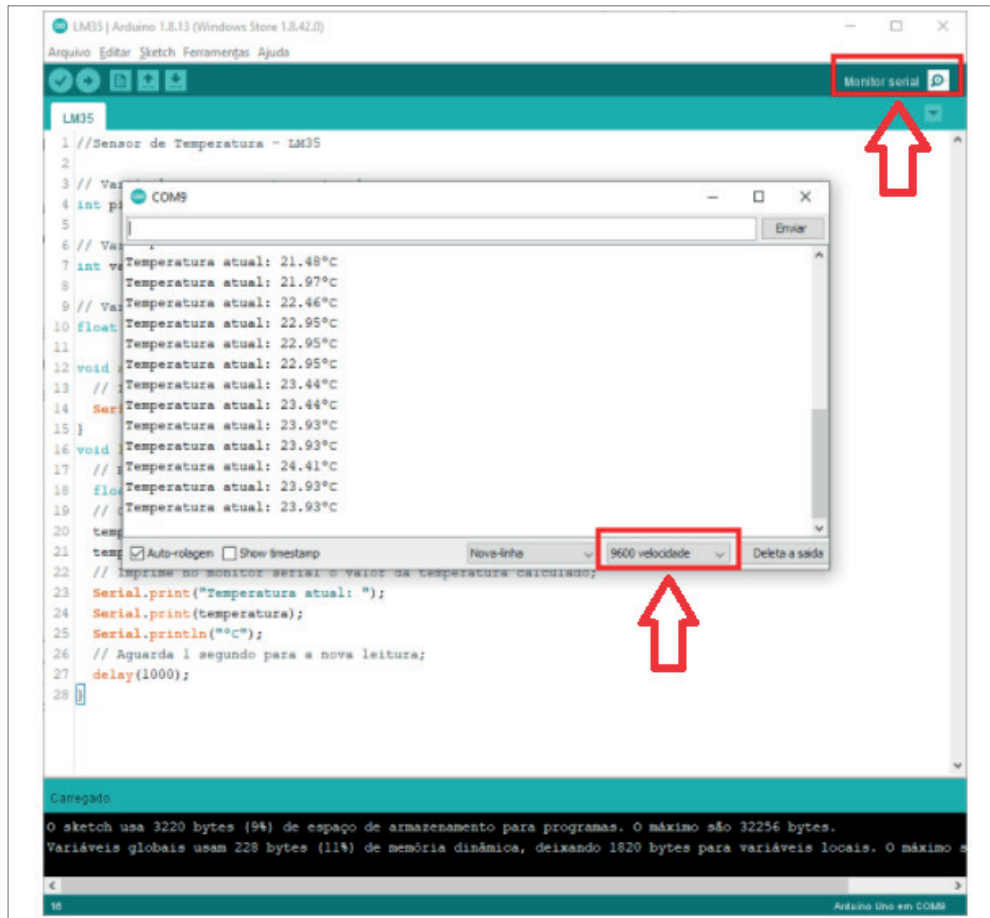
void setup() {
  Serial.begin(9600); /* Inicializa a comunicação serial
*/
  pinMode(LM35, INPUT); /* Define o pino A0 como ENTRADA. */
}

/* Função que será executada continuamente */
void loop() {
  /* Determina a temperatura a partir da tensão obtida do
LM35 */
  Temperatura = (float(analogRead(LM35)) * 5 / (1023)) /
0.01;
  Serial.print("Temperatura: ");
  Serial.print(Temperatura);
  Serial.println("°C");
  delay(2000);
}
```

Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verify** (botão com sinal de tique) para verificar se não há erros de sintaxe. Estando o código correto, o próximo passo é realizar a transferência do programa para o Arduino. Para tal, pressione o botão **Upload** (botão com uma seta apontando para a direita).

Após a transferência do programa para o Arduino, você deverá acompanhar os valores de temperatura através do monitor serial, para isto, enquanto o Arduino estiver ligado, clique sobre o botão localizado no alto e à direita da janela do Software Arduino IDE, como demonstra a figura 4.

Figura 4 - Monitor Serial do software Arduino IDE



Atenção! Para que as informações do sensor sejam apresentadas corretamente na tela, é necessário configurar a velocidade de leitura, verifique a velocidade indicada (figura 4), a qual deve ser igual à velocidade informada no código (quadro 1), dentro da função **setup**, nesse caso 9600 baud.



## Para Saber Mais...

O termo baud origina-se de Émile Baudot, francês que inventou o código telegráfico, e corresponde à unidade de medida de velocidade de sinalização (em telecomunicações) ou taxa de transmissão de dados entre computadores, ou seja, a quantidade de símbolos que são enviados por segundo.

## ii. Linguagem de programação por blocos

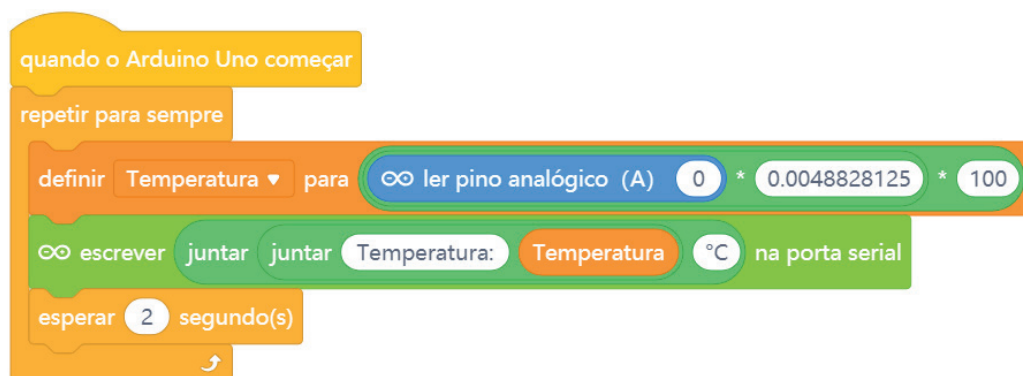
Outra forma de programar o sensor de temperatura é por meio da linguagem de programação que utiliza blocos de funções prontas, os quais representam comandos de programação. Vamos utilizar o software mBlock.

Para conectar o mBlock ao Arduino, você deve clicar no ícone **Adicionar**, localizado no campo **Dispositivos**, e selecionar o Arduino, na biblioteca de dispositivos do mBlock, clicando, na sequência, no botão **OK**.

Nesta programação, utilizaremos variáveis que auxiliarão na estrutura do nosso programa. Para recordar como criar uma variável, consulte a **Aula 05 - Softwares Arduino IDE e mBlock**.

Monte os blocos, arrastando e soltando, de acordo com a programação do projeto, como mostra a figura 5.

Figura 5 - Programação em blocos para funcionamento do sensor de temperatura LM35

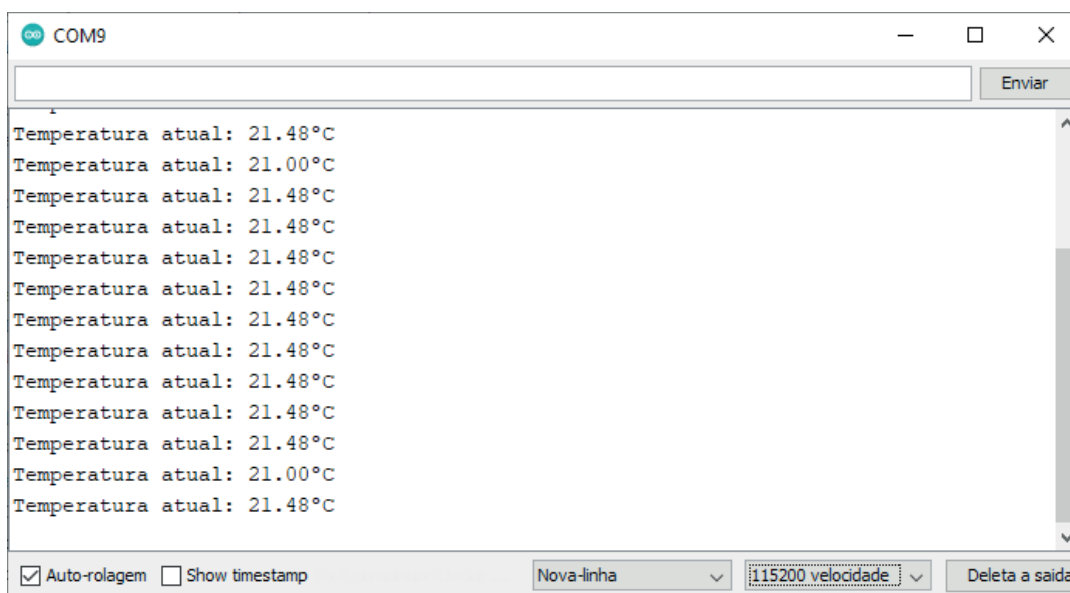


Assim que os blocos estiverem montados, clique no botão **Conectar** para iniciar a comunicação entre o software mBlock com a placa de Arduino Uno. Ao clicar sobre o botão Conectar, aparecerá um *Tooltip* solicitando a confirmação da conexão entre os dois dispositivos.

Uma vez realizada a conexão entre os dispositivos, será ativado, na interface do mBlock, o botão **Upload**, o qual ao ser clicado, o software verificará se não há erros na estrutura do programa e, então, compilará para enviar o programa à placa Arduino.

Após carregar o programa é hora de observar os valores da temperatura fornecido pelo sensor, para isto feche o Software mBlock e abra o Software Arduino IDE. A seguir, clique no **Monitor serial** (figura 4) e confira a velocidade indicada no rodapé do *tooltip* que se abre, a qual deve registrar o valor de 115200 baud, como mostra a figura 6. Esta velocidade é padrão no software mBlock. Caso, a velocidade esteja com valor diferente a este padrão, faça o ajuste, clicando na seta ao lado do box que indica a velocidade.

Figura 6 - Monitor Serial do Software Arduino IDE



### Desafios:

- i. Experimente colocar o dedo no sensor e veja o que acontece com os valores de temperatura obtidos na tela.
- ii. Integre neste projeto, a partir dos conhecimentos desta aula, um buzzer para criar um sistema de alarme, que dispare ao alcançar um determinado valor temperatura.
- iii. Altere sua programação para converter os valores de temperatura para outras escalas (Fahrenheit e Kelvin). Dica: Pesquise por fórmulas matemáticas que convertam uma temperatura na escala Celsius para essas outras escalas e incorpore-as na programação.



E se... ?

O projeto não funcionar, se atente a alguns dos possíveis erros:

- Verifique se os jumpers estão na mesma coluna dos terminais dos componentes, fazendo assim a conexão;
- Verifique se os jumpers estão ligados nos pinos corretos no Arduino;
- Verifique se a programação está adequada a cada porta digital;
- Verifique se a velocidade de leitura no monitor serial está igual à velocidade informada no código de programação.

### 3. Feedback e Finalização (15min):

**a.** Confira, compartilhando seu projeto com os demais colegas, se o objetivo foi alcançado.

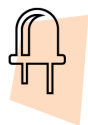
**b.** No projeto desenvolvido, você conseguiu verificar se a temperatura medida corresponde à realidade?

**c.** Reflita se as seguintes situações ocorreram:

**i.** Colaboração e Cooperação: você e os membros de sua equipe interagiram entre si, compartilhando ideias que promoveram a aprendizagem e o desenvolvimento deste projeto?

**ii.** Pensamento Crítico e Resolução de Problemas: você conseguiu identificar os problemas, analisar informações e tomar decisões de modo a contribuir para o projeto desenvolvido?

**d.** Organize e guarde os materiais utilizados no kit de robótica.



## Videotutorial

Com o intuito de auxiliar na montagem e na programação desta aula, apresentamos um videotutorial, disponível em:



<https://rebrand.ly/a29robotica>

Acesse, também, pelo QRCode:

