

Actividad 1: Medición de temperatura

NAME: _____ ID: _____
NAME: _____ ID: _____
NAME: _____ ID: _____
NAME: _____ ID: _____
NAME: _____ ID: _____

PRERREQUISITOS: Para esta práctica es necesario tener instalado un ambiente de arduino, si no tienes el IDE instalado, [aquí](#) hay un tutorial sobre como hacerlo. Es necesario que tengan al menos un arduino y una compu con el IDE por equipo.

INSTRUCCIONES: Trabaja en equipo en esta actividad. Siempre que se solicite, anota tu procedimiento, incluyendo fórmulas, diagramas y cálculos, así como las unidades correspondientes en las respuestas que así lo requieran. Si ya tienes experiencia en esta área apoya a tus compañeros de equipo, el objetivo no es entregar la práctica, sino que todo el equipo sea capaz de conectar y utilizar sensores correctamente.

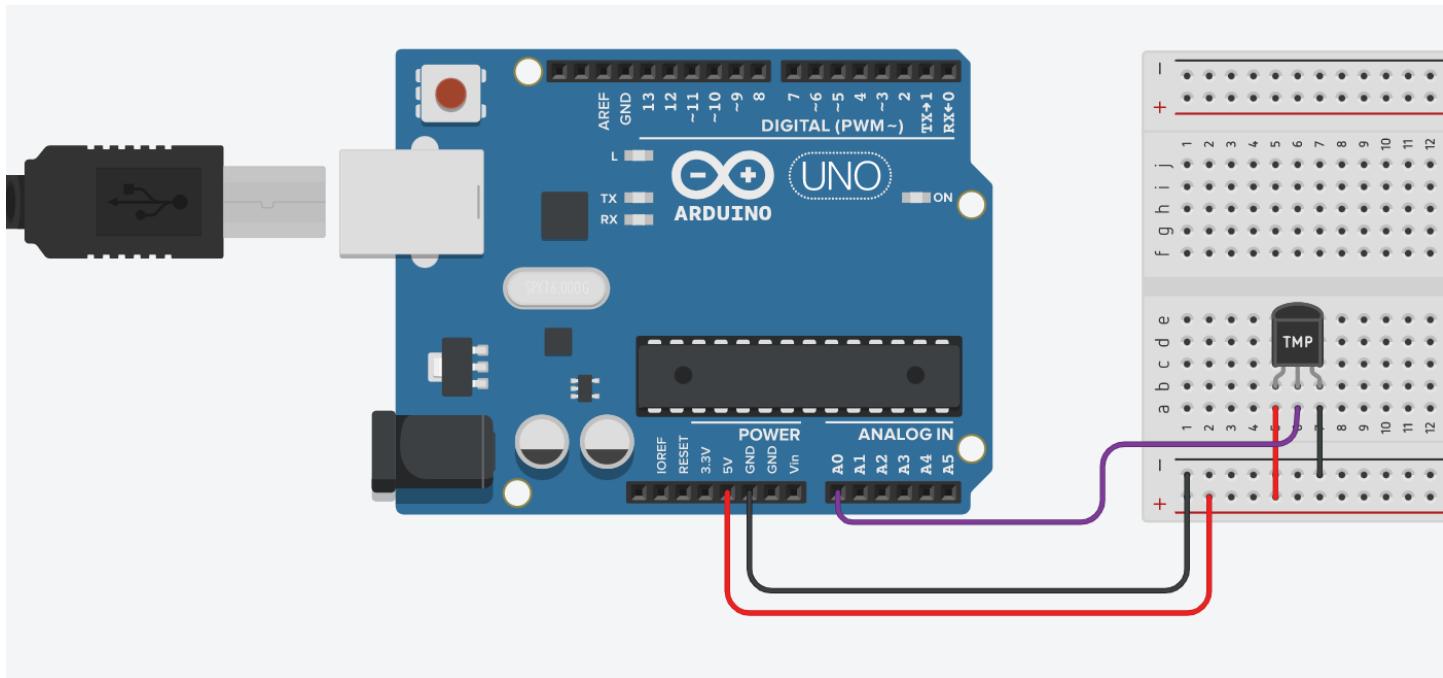
OBJETIVO: Realizar la lectura de datos de temperatura (Aplicación invernaderos)

MATERIAL NECESARIO

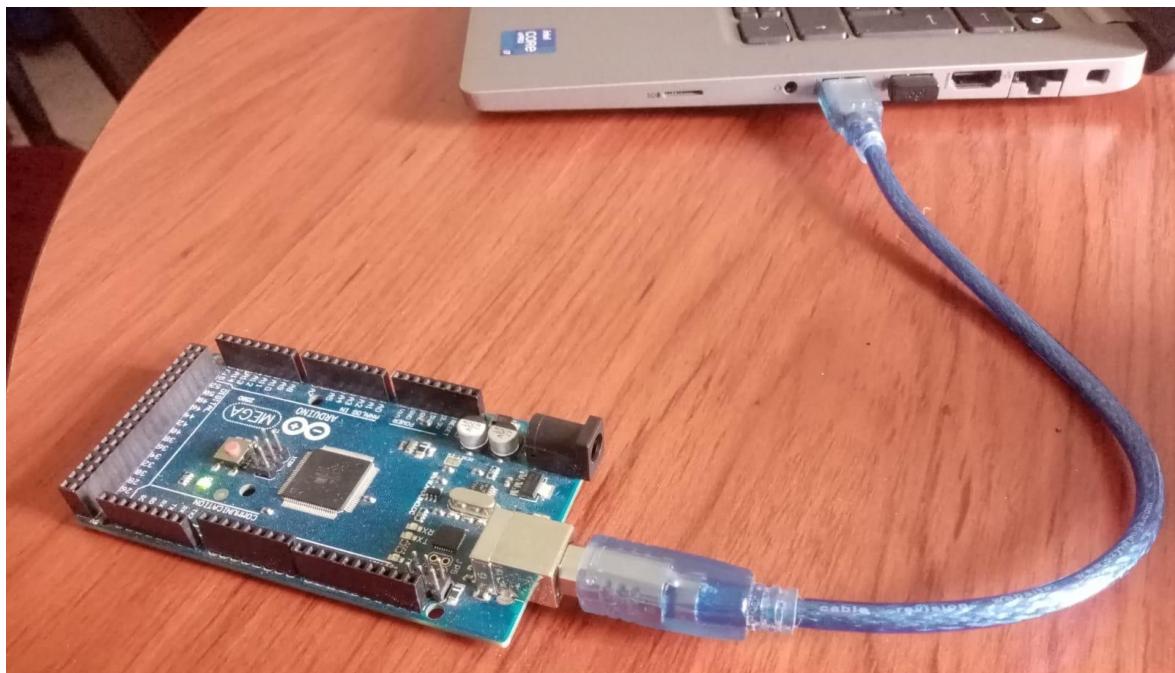
- 1 Arduino UNO
- 1 sensor de temperatura analógico
 - (TMP36) recomendado o LM35 (Opcional)
- Cables jumper

CIRCUITO:

Conecte los componentes como se indica en la siguiente imagen



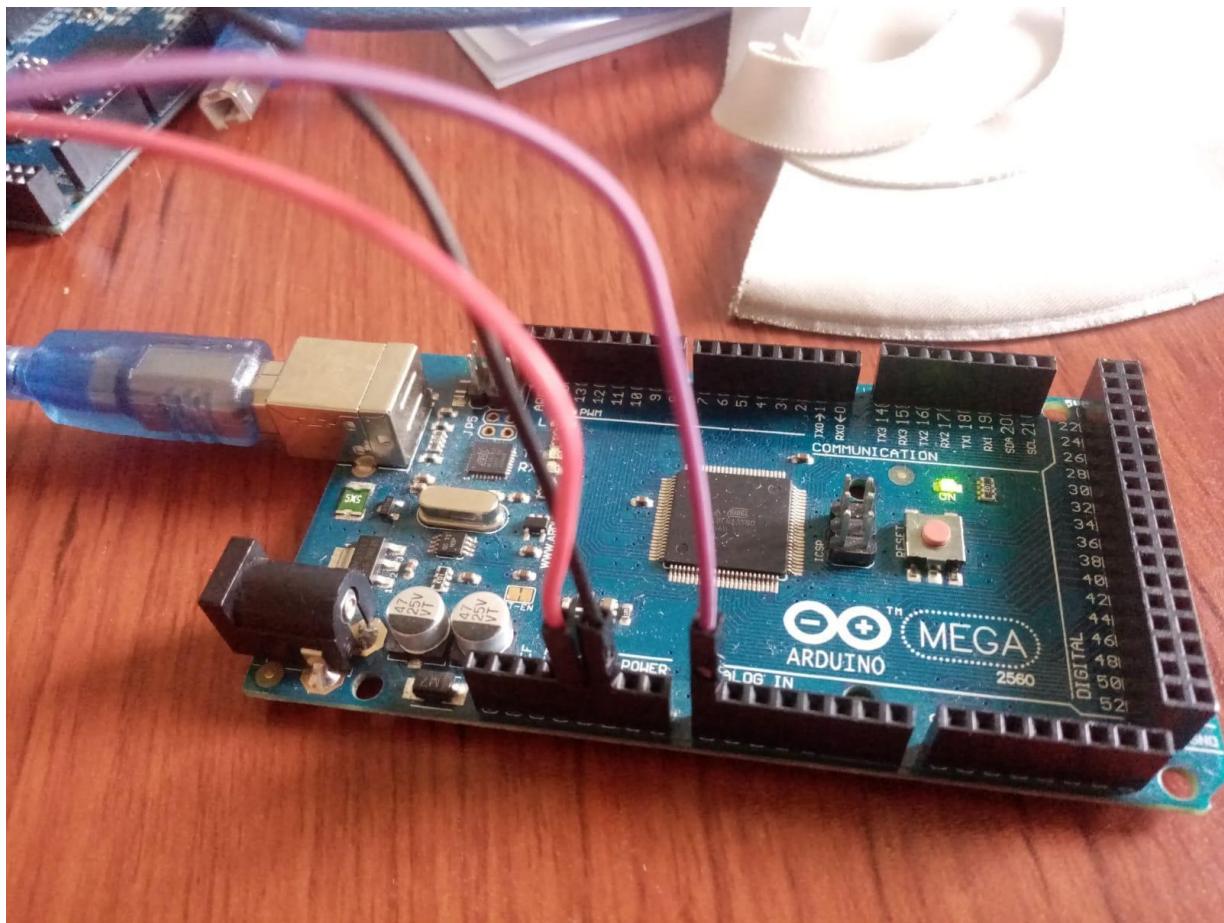
Así se ve una tarjeta arduino conectada a usb



este es un sensor de temperatura:



Así se ve todo conectado:



Programa Base

Utilice el siguiente programa como base para empezar a desarrollar su práctica en el IDE de Arduino:

```
//Sensor de temperatura usando el LM35 o TMP36 junto con un LCD
//Definiciones
int pin_sensor = A0; // Define el pin analogico del LM35
int valorA0;      //Variable entera para leer el valor de A0

void setup() {
    Serial.begin(9600); // Inicia la comunicación con el monitor serial de Arduino
}

//Función que será ejecutada continuamente
void loop() {
    //Lectura del Pin A0 (ADC)
    valorA0=analogRead(pin_sensor); // Almacena el valor entero.
    Serial.println(valorA0);
    delay(1000); //Imprime el valor cada segundo
}
```

Marco teórico.

Discutan en equipo y respondan las siguientes preguntas.

1. ¿Qué es una señal analógica?
2. ¿Cuál es la principal diferencia entre una señal analógica y una señal digital?
3. En la tarjeta Arduino Uno, ¿qué pines son los pines que se usan para entradas analógicas? Busca e incluye una imagen indicando tu respuesta.
4. En Arduino, ¿cuál es la instrucción para leer el valor actual de un pin analógico? (escribe la sola instrucción con sus parámetros)
5. ¿Cuál es la instrucción para leer el valor analógico en el pin analógico A3? (escribe la sola instrucción con sus parámetros)
6. La función *analogRead()*, ¿qué valor regresa? ¿Cuáles son sus características?
7. A fin de conocer el valor real de voltaje en un pin analógico, ¿qué conversión debe hacerse? Anota la fórmula correspondiente.
8. Explica que es un sensor de temperatura.

Escenario.

A fin de optimizar la producción de jitomate, se deben controlar factores como la cantidad de agua disponible en la tierra para modular la absorción de nutrientes, así como la temperatura para tener cierto control sobre los procesos químicos propios del crecimiento de la planta. La temperatura ideal para el crecimiento y mantenimiento de este tipo de plantas es de 20 a 22 °C, durante el día, aunque durante la floración se recomienda en promedio 18 °C.



Figura 1. Producción de jitomate en un invernadero.

En cierto invernadero se busca monitorear la temperatura de su interior para controlar la producción, y como ingeniero se te ha pedido crear un módulo de monitoreo de la misma. Para la selección del sensor, hay muchas posibilidades, como el *LM35* o cualquiera de la serie *TMP35* – 37. En este caso, tenemos el *TMP36* disponible, que tiene una respuesta lineal, regresando incrementos de $10mV$ por $^{\circ}C$, y trabaja en el rango desde los $-40^{\circ}C$ hasta los $+125^{\circ}C$. Más particularmente, la relación entre la señal de salida de este sensor (en Volts) y la temperatura sensada, está dada por:

$$v = \frac{T}{100} + 0.5$$

En donde v es el voltaje que regresa el sensor (en Volts) y T es la temperatura registrada (en $^{\circ}C$).

El sistema que debes crear es un circuito para leer la temperatura y mostrar su valor en pantalla, y al mismo tiempo encender un indicador verde si la temperatura está en el rango adecuado para la planta, y rojo si la temperatura se sale de este rango, y mostrar mensajes adecuados. En la pantalla debería verse algo como lo siguiente (considera delays de 1s entre cada lectura del sensor):

```
Current TEMPERATURE: 21.5 oC - Optimum conditions
Current TEMPERATURE: 20.8 oC - Optimum conditions
Current TEMPERATURE: 20.9 oC - Optimum conditions
```

```
Current TEMPERATURE: 21.2 oC - Optimum conditions
Current TEMPERATURE: 22.3 oC - Warning! Out of range...
Current TEMPERATURE: 22.8 oC - Warning! Out of range...
Current TEMPERATURE: 21.1 oC - Optimum conditions
Current TEMPERATURE: 20.2 oC - Optimum conditions
Current TEMPERATURE: 19.4 oC - Warning! Out of range...
Etc.
```

Integración de Data Set

Una vez que han llegado a este punto ahora es necesario armar un set de datos con la información recolectada.

Implementa un script usando pySerial para que en vez de mostrar los datos en la pantalla de arduino, los capture en un archivo de texto.

instala pyserial con:

```
pip install pyserial
```

crea un archivo con el siguiente código para poder ver las señales a cualquier sensor.

```
import serial
from time import sleep

#puerto default /tty/ACM0
ser1 = serial.Serial(<PUERTO DEL SENSOR> , 9600)
sleep(3)

while(1):
    arduinoData = ser1.readline()
    print(arduinoData)
```

Nota: si te da el siguiente error, consulta la sección de troubleshooting en el [tutorial](#)

```
avrdude: ser_open(): can't open device "/dev/ttyACM0": Permission denied
```

Ahora modifica el código para que en vez de enviarlo a la pantalla de manera independiente, integre la fecha a cada renglón y lo guarde en un archivo de texto.

para obtener la fecha actual usa la biblioteca [datetime](#):

```
from datetime import datetime
```

Ahora convierte los datos del sensor a un formato numérico de python con un cast a int, con esto ahora puedes utilizar el dato para usar en python. Genere el estatus de Conditions y selecciona un rango de temperatura adecuada a la aplicación, donde queremos que valores de temperatura dentro del rango adecuado se graben como Optimum y valores fuera del rango queden como Warning.

Integra todos los datos en un data frame de pandas y salva el dataframe a un csv usando [df.to_csv](#)

```
import pandas as pd
```

Ahora genera una gráfica de la temperatura contra el paso del tiempo usando [matplotlib](#)

```
import matplotlib.pyplot as plt
```

Al final se espera que generen un data set que se vea aproximadamente como este:

Date	Current TEMPERATURE	Conditions
19-03-2022	21.2	Optimum
19-03-2022	22.3	Warning

Discusión Final.

Discutan en equipo y respondan las siguientes preguntas.

9. ¿Para qué serviría tener un set de datos como el que acabas de armar?

10. ¿Qué otra información podría ser interesante conseguir además de la temperatura en este escenario?

11. ¿Qué otras aplicaciones o dataset podrían generar integrando otros sensores?

