

Feast or Famine Terrarium Project (Resource Exchange)

Vicki V. May (Professor)

Professor of Engineering at Dartmouth.

Samuel S Streeter (Graduate Student)

Sara Vannah

Feast or Famine Terrarium Project (Resource Exchange)

Overview

Program Description: Through a project funded by the Science Education Partnership Award of the National Institute of General Medical Sciences, a team of faculty, staff, and graduate students from Dartmouth as well as educators from a local science museum are collaborating with middle-school STEM teachers from rural schools to develop project-based curricula.

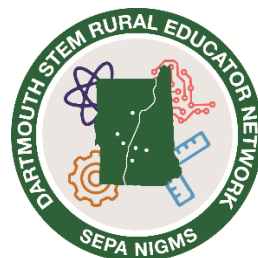
The Feast and Famine curriculum focuses on having middle school students build a terrarium, plant and care for edible seeds/plants, design experiments, design and build measurement tools, and assess how different factors affect the plants in their terraria. This is an interdisciplinary project that incorporates environmental and engineering concepts. **Engineering concepts** include the use of laser-cutters to create the acrylic terrariums (CAD files available but students can also use 2-liter bottles) and the design and use of Arduino-based tools to measure temperature, humidity, illuminance, and more. In addition, students will use time-lapse photography to monitor plant growth.

Grade Level: 7th grade

Time: The curriculum is designed to be run over 4 to 6 weeks but can be adjusted as needed.

Vicki May¹, PhD, Roger Sloboda², PhD, Michele Tine³, PhD, Samuel Streeter¹, David Clemens-Sewall¹, Sara Vannah⁴, and Genevieve Goebel⁴

¹Thayer School of Engineering, ²Biological Sciences, ³Sociology, and ⁴Guarini School of Graduate & Advanced Studies
Dartmouth College



<https://sepa.host.dartmouth.edu/>

Learning Objectives

Next Generation Science Standards:

- MS-LS1-5. Construct a scientific explanation based on evidence for how environmental factors influence the growth of organisms.
- MS-ETS1-1. Define the criteria and constraints of a design problem with sufficient precision to ensure a successful solution.
- MS-ETS1-2. Evaluate competing design solutions using a systemic process to determine how well they meet the criteria and constraints of the problem.
- MS-ETS1-3. Analyze data from tests to determine similarities and differences among several design solutions.
- MS-ETS1-4. Develop a model to generate data for iterative testing and modifications of a proposed tool.

Supplies

- Terrariums (CAD plans or use 2L bottles)
- Soil, sand, pebbles, fertilizer
- Seeds: beet, basil, and brassica
- Grow light
- Arduino board
- USB cable
- MicroSD card reader
- Sensors: moisture, light, humidity
- Time-lapse camera
- Breadboard
- Jumper wires

Students Design and Build Arduino-Based Tools

Students use the engineering design process to design, build, modify, and test Arduino-Based Tools.

Be an Engineer

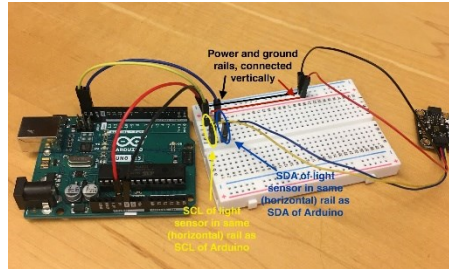


- Ask** How do sensors, Arduino, and Python work? How can you use these tools?
- Define** What do you need to measure? How often do you need to measure it?
- Wonder** What are the possibilities? How many ideas can you come up with?
- Build** What measurement tools can you build? Can you combine several tools?
- Test** Do your tools work? How can you interpret the data? How can you improve your tools?

Students Design and Code Tools/Sensors to take Measurements using Arduino

- Lesson 1: Intro to Arduino Boards and Coding
- Lesson 2: Building your first Sensor
- Lesson 3: Recording Data over Time
- Lesson 4: Integrating Multiple Sensors with a Real-Time Clock
- Lesson 5: Collecting and Interpreting Time-Lapse Imagery

Link to lessons:
https://drive.google.com/drive/folders/1-BOs_spdqf7gszoyq_qXdqrmfp004vbl?usp=sharing



```

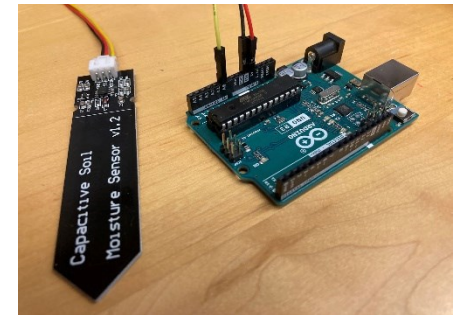
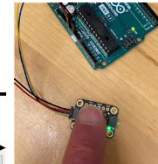
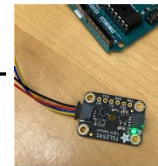
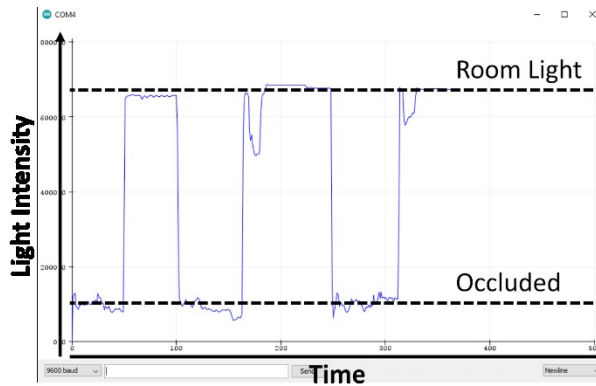
1 //
2 //
3 // Turn on LED on for one second, then off for one second, repeatedly.
4 //
5 // This example code is in the public domain.
6 //
7 // http://www.arduino.cc/en/tutorial/basics
8 //
9 //
10 //
11 // define statements make code more legible by letting us define things
12 // names in the code - eg. "int ledPin" is the integer 13
13 #define LED_PIN 13
14 //
15 // the setup function runs once when you press reset or power the board
16 void setup() {
17   // initialize digital pin LED_BUILTIN as an output.
18   pinMode(LED_PIN, OUTPUT);
19 }
20 //
21 // the loop function runs over and over again forever
22 void loop() {
23   digitalWrite(LED_PIN, HIGH); // turn the LED on (HIGH is the voltage level)
24   delay(1000); // wait for a second
25   digitalWrite(LED_PIN, LOW); // turn the LED off by making the voltage LOW
26   delay(1000); // wait for a second
27 }
    
```

Comments in grey or preceded by "//" or sandwiched by "/* */" are meant to inform the user of each section's purpose

Definitions and imports go up here

Setup() prepares 'ingredients' of program

Loop() includes instructions and tells microcontroller to complete tasks repeatedly, as frequently as specified



Time-Lapse Videos

