Three Blinking Lights

Academic Integrity Guidelines
This is an individual assignment, and all code you write should be your own. You should not use any resources other than those linked to in the assignment, or provided by the professor or TAs. You should not look at anyone else's code, or show anyone your code.

Learning Goals:
- Get familiar with the Arduino programming environment.
- Learn how to build a circuit to connect external components to the Arduino.
- Learn how to program the Arduino to control external components.
- Bonus: Pretty blinking lights.

As working with microcontrollers and electronics is new to most of you, don't worry if some of this seems weird or confusing. Please attend the lab during your scheduled lab time, or any of the office hours, in order to get help setting everything up.

Prelab: (Things to do on your own to understand how to do the actual lab!)
Get a blinking LED on Pin 13. (Note: Pin 13 controls both the onboard LED, and the external pin 13. If you do this correctly, your onboard and external LED should blink at the same time.)

You will first need to get the Arduino IDE set up on your machine. You can download it here:
- https://www.arduino.cc/en/software.

There are plenty of tutorials for setting up Arduinos on the Arduino website.

You can follow one of the many online tutorials on how to do this, such as:
- https://docs.arduino.cc/built-in-examples/digital/BlinkWithoutDelay/

Notes:
- A good way to approach this is to get your onboard LED blinking first, and then connect an external LED.
- The Arduino IDE comes with sample code to get your LED blinking.

Lab 1: Due by end of lab time Monday 09/8/2025

Materials
- 1x Arduino
- 1x Breadboard
- 1x Red LED
- 1x Green LED
- 2x 220 Ohm resistors
- Many wires

Use the onboard LED on pin 13, a red LED on a second digital pin of your choice and a green LED on a third digital pin of your choice.

You are to have these three LED's cycle so only one LED is on at any given point in time. First, your onboard LED should turn on, then it should turn off and your red LED should turn on, then your red LED should turn off and your green LED should turn on, then your green LED should turn off and the

onboard LED should turn on, then your red LED should be the only one on, then your green LED should be the only one on, then your onboard LED should be the only one on, then your red LED should be the only one on, etc . . .

Each LED should be on for about a second (long enough to show that only one LED is on at any time but not too long that the demo takes forever).

The digital pins should be labeled 0-13 on your board.

You will NOT receive full credit if you use delay() anywhere in your code. This is a 300-level Computer Science course, and a certain level of programming maturity is expected from the students.

- Using the millis() function (https://www.arduino.cc/reference/en/language/functions/time/millis/) is expected for this. Note: millis() can still be used incorrectly! Beware of how you are using it! Note: putting a millis() statement inside a loop statement is the same as using a delay()!
- You can also check this article Why Use millis() Instead of delay()? (https://www.norwegiancreations.com/2017/09/arduino-tutorial-using-millis-instead-of-delay/).
- https://docs.arduino.cc/built-in-examples/digital/BlinkWithoutDelay/
- https://forum.arduino.cc/t/using-millis-for-timing-a-beginners-guide/483573

Each off-board LED should be connected to a 220 Ohm resistor. Note that this will mean you have to connect two separate LEDs to the ground pin. We recommend using a breadboard for this.

If you wish, you may wish to connect a third off-board LED to the Digital Pin 13. If so, pick a different color than Red or Green for the LED. Using a third off-board LED is not a requirement for the lab (but it is much easier to see a third off-board LED blinking than the on-board LED).

During your demo, expect to be asked questions about your design. Such questions might be things like:

- What would you change if we wanted the green LED to be on pin 4 or on pin 11?
- What would you change if we wanted each LED to be on for 20 seconds?
- What would you change if we wanted to use a yellow LED instead of the onboard LED?
- What would you change if we wanted all LED's to be off for a second before turning the next LED on?
- Why are we not allowed to use delay() to make this lab function? What functionality is lost when delay() is used?
- What might happen if we didn't use a resistor when wiring each LED?
- What might happen if we used a 10 Ohm resistor or a 10K Ohm resistor when wiring each LED?
- If the TA were to remove a wire from your design, how would you debug the situation to determine what actions are needed to fix it?
- How would you determine if a wire is physically broken inside of its plastic insulation?

Properly completing the above tasks will only earn 85% of the points for the lab. To earn the final 15% of the points you must add some additional functionality to the lab. This additional functionality:

- Must not change or interfere with the above required functionality.
- Must be somewhat similar to the required functionality for this lab, BUT different from the required functionality.
- This additional functionality could be something like adding on additional LEDS that blink for a different amount of time than used for the required functionality. These additional LEDs could blink using some different pattern or even blinking for a variable amounts of time.

Due by Lab Time Week 3: Have a TA check out your lab project by the end of your scheduled lab on Monday 09/8/25, **AND** submit the code via Gradescope **prior** to the demonstration of your lab.

Import notes concerning the submission of your Arduino Code for the Lab:

1. Your code must be written as an .ino code file using C/C++ as the base language. Other programming languages are not accepted for CS 362.

2. If you fail to submit your code to Gradescope before you demonstrate your project to the TA, the TA will not have an entry in Gradescope to record your results. Thus, your demo would be pointless and would require you to demonstrate your lab again.

3. Updating your Gradescope submission after you demonstrate your project to TA will cause another entry in the Gradescope system which will cause any points from your demonstration to be lost. Thus requiring you to demonstrate your lab again.

4. Failure to follow these points will result in a large deduction of the points for your lab.

5. If possible, just prior to your demonstration add a comment at the end of your Header Comments with the Date, Time and Name of the TA to whom you are demonstrating. If a question comes up regarding the grading or demonstration of your lab, having that information can mean the difference between successfully resolving the question or not getting it resolved.

What should I include with my .ino Code File?
As with any code file, it should be written in Good Coding Style: in a manner that will help other people read and understand the intent, purpose, operation of the code. There will be points assigned for the following items in your code. So if you indent to earn the maximum amount of points, your code must include:

- Naming the .ino file with your NetId and Lab Number
  - I.E. something like: emccartyLab1.ino
  - where “emccarty” is replaced with your own NetID
  - your NetID is the part of your UIC email address before the “@uic.edu”
- Header Comments (including the following)
  - // FirstName LastName, UIN and NetID
  - // Lab Number and Title of the Lab
  - // Description – explaining what is this code intended to do
  - // Include any assumptions you may have made, what do you expect from the hardware, pinouts, particular arduino versions, etc.
  - // References - where did you find code snippets, ideas, inspirations? if no references used say: "no references used"
- Code is well documented/formatted with inline comments, indentations, descriptive variable names, and properly uses functions
- Actual code - the functions in the cpp/ino file

About a third of the total points for the lab will based on the code file submitted. However; if you just submit working code without doing a demo, you will only get zero points.

Academic Integrity Guidelines for this lab: You are allowed to consult resources linked from this lab write up, and anything provided by the professor or TAs via blackboard, as well as any resources you need in order to get the Arduino software and hardware working with your computer. You should NOT look up any of the code you need to get the LEDs to turn on and off as described, you should be able to figure this out yourself from the materials provided. This is an individual assignment, and you should not look at anyone else's code, or show anyone your code.

Lab Late Policy

- Lateness is determined by the time the lab is demonstrated, not when the .ino file is submitted.
- Labs that are not demonstrated get a score of 0.
    - -50% for no demonstration
    - -50% for being late
- Any lab demonstrated after 5:50pm on the day when the lab is due is considered late.
- Late Deadline 1: demonstrated by the Thursday after the Lab Due date
    - Demonstrated after the end of Lab Time on 9/8, but before 6:00pm on Thursday 9/11
    - 25% Penalty
- Late Deadline 2: up to 1 week late
    - Demonstrated between Friday 9/12 but before 6:00pm Monday 09/15/2025.
    - 50% Penalty
- Labs Demonstrated after 1 week late receive a score of 0 (as if they were never demonstrated).