```
///////////////////////////////////////////////
/* HELEN"S PSEUDO CODE

*** SET-UP ***
Set up & establish arduino connection with all components
    - mouse eye 1, green LED, OUTPUT for low tones
    - mouse eye 2, yellow LED, OUTPUT for high tones
    - calibrator, LED, OUTPUT will display light during millis calibration count
    - photosensor, analog INPUT, reads light
    - piezo, digital OUPUT, sound

Set up variables to hold high and low
(I found this brilliant calibration method in Scott's code. Previously, I had attempted to
hard code sensorHigh and sensorLow based on values found in serial printouts.)

   initialize sensorValue, which is a variable for storing the current photosensor value

   initialize sensorLow
      SensorLow is set to a value near the upper limit of the photosensor serial reads. By
      purposely starting the variable at a high value we can be sure to overwrite it with an
      appropriate low value during the calibration loop.
      After calibration, this variable will hold the lower light limit for our map function.

    initialize sensorHigh to 0
      By starting at 0 we can be sure to overwrite it with an appropriate high value during
      the calibration loop.
      After calibration, this variable will hold the upper light limit for our map function.

   initialize eyeDiff, which is a variable that stores the difference b/t sensorHigh &
   sensorLow
      eyeDiff/2 will be used to as a switch b/t the mouse's green (low tone range) and yellow
      (high tone range) eyes

Run calibration for 5s using mills()
  reset sensorHigh value
     Each time a value is recorded that is higher than the current state of sensorHigh,
     rewrite sensorHigh with the new high value.

  reset sensorLow value
      Each time a value is recorded that is lower than the current state of sensorLow,
      rewrite sensorLow with the new low value.

  turn off calibrator LED to signal the end of the mills calibration period

  calculate eyeDiff
      (sensorHigh-sensorLow)/2

Check the current values of sensorHigh, sensorLow, & eyeDiff via Serial.println()


*** LOOP ***
Read the input from the photosensor and assign it to sensorValue

Output light based on sensorValue data
- green mouse eye will light when low tones are read
- yellow mouse eye will light when high tones are read

Translate the sensorValue data to tones values using map()
- store the mapped tone values in a new variable called pitch

Output sounds to piezo based on pitch values


END PSEUDO CODE */
///////////////////////////////////////////////
```