Assignment 2

- The hardware implementation

  Components: Arduino board, breadboard, ultrasonic distance sensor, buzzer, LED, resistor, wires.

  Connections:

  - I connected the breadboard to GND and 5V of Arduino.
  - I connected the ultrasonic distance sensor's pins this way: Vcc to 5V, Trig to D9, Echo to D10, Gnd to Gnd.
  - I connected the buzzer's pins: - to Gnd, + to D11.
  - I connected the LED's anode to D13 and the cathode to Gnd, using an appropriate resistor to limit the current (the resistor has 200 Ohms).

- The software implementation

  I declared variables for the pins of the LED, buzzer and ultrasonic distance sensor (trigPin, echoPin, brakeLedPin, buzzerPin) and for the computation of the detected obstacle's distance (duration, distanceCm).

  After that, in the setup function, I set the pins' mode: echoPin as input and trigPin, brakeLedPin, buzzerPin as outputs.

  In the loop function I set the trigPin to LOW, we delay the code by 2 microseconds to make sure that the trigPin is clear; then I set the trigPin to HIGH for 10 microseconds to generate the ultrasound wave. After that, the trigPin is set to LOW again.
  Then we read the echoPin with the pulseIn function which returns the wave travel time in microseconds.
  For getting the distance in cm I will multiply the duration by 0.034 and divide it by 2.

If the obstacle is closer than 50 cm, the red LED will receive a HIGH signal, simulating the brakes' engagement. To do the buzzer's beep sound to alert the driver I sent 1KHz to the buzzer for 500ms, then I stopped the sound.

At the end of the loop, I delayed the program by 100 ms.

The software implementation is found at the following link: https://www.tinkercad.com/things/2xW6p5yosij-assignment-2-rtcs-ionescu-natalia-cen4ha/editel?returnTo=%2Fdashboard&sharecode=XO3RDkH0VL95LeMC_PJvXZTjcGq8qbJzP_5Vt4KlT_A

- Personal information

  For the brakes' engagement simulation I used a LED on port 13, for which I sent a HIGH signal .