Discover Engineering!
Workshop to make an *Ultrasonic Sight Device* to aid a person with a vision impairment

Code for the Ultrasonic Sight Device:

The code to run the ultrasonic sight device is available at:

https://github.com/mitchbryson/ultrasonic_sight

Program Code:

```
/*
Ultrasonice Sight Device: uses an Arduino Uno connected to a Sonar Range Sensor and Buzzer
This code will produce beeps that get faster as an object gets closer.
*/

// Define variables for how the code works
int const max_distance = 100;

// Define pins for ultrasonic and buzzer
#define TRIGGERPIN 10
#define ECHOPIN 9
#define BUZZERPIN 2

void setup() {
  pinMode(TRIGGERPIN, OUTPUT); // Pulses will be sent out of the Arduino
  pinMode(ECHOPIN, INPUT); // Echo pulses will be received in to the Arduino
  pinMode(BUZZERPIN, OUTPUT); // Arduino controls whether the buzzer is on or off
}

// beep_rate: This function controls the rate of beeping based on distance
void beep_rate(int distance) {
  int num_beeps = 1;
  int beep_sep = distance;
  if (distance < 20) {
    beep_sep = 20;
  }
  for (int i = 0; i < num_beeps; i++) {
    digitalWrite(BUZZERPIN, HIGH);
    delay(distance);
    digitalWrite(BUZZERPIN, LOW);
    delay(distance);
  }
}

// This function is called repeated in a loop to control the whole system
void loop() {
  // Duration will be the input pulse width and distance will be the distance to the obstacle in centimeters

  // Send an output pulse to the sonar sensor (1ms wide)
  digitalWrite(TRIGGERPIN, HIGH);
  delay(1); // wait 1ms
  digitalWrite(TRIGGERPIN, LOW);

  // Measure the time taken to receive the pulse back
  // Will be "0" or "-1" if no pulse is received
  int duration = pulseIn(ECHOPIN, HIGH);

  // Formula for the distance is half the duration, then divided by 29.1
  // to get distance in cm
  int distance = (duration/2) / 29.1;

  // if distance less than 0.5 meter and more than 0 (0 or less means over range)
  if (distance <= max_distance && distance >= 0) {
    digitalWrite(BUZZERPIN, HIGH); // Buzz
    //beep_rate(distance); // Alternative way of beeping
  } else {
    // Don't buzz
    digitalWrite(BUZZERPIN, LOW);
  }

  // Wait for 60ms until running the code again in a loop
  delay(60);

}
```

Coding Instructions:

- **Setting up the Arduino IDE:**
  - Go to the web address above and click on the link to the code file called "ultrasonic_sight001.ino"
    - This contains the code for the device: we will compile this code and upload it to your Arduino using the Arduino software on your laptop/computer and the USB cable
  - Open up the software/program "Arduino" on your laptop/computer (sometimes called the "Arduino IDE":
    - If you don't have this installed already, you can download it for free from https://www.arduino.cc/en/software
  - Inside the Arduino IDE software, click on the menu "Tools" then "Boards" and select "Arduino Uno"
    - This will tell the software the type of Arduino we are programming
- **Compiling and running the code:**
  - Go to the menu "File" then "New" to start a new Arduino sketch
  - Delete the contents of the new file then copy and paste the code into this file
  - Click on the "Verify/Compile" button (the big tick inside a small circle, located up the top left of the Arduino IDE window) to compile the code.
    - If there are any code errors, and error message in orange writing will appear at the bottom of the program window
  - Connect the Arduino Uno to your laptop/computer using the USB cable
  - The click on the "Upload" button (the right-facing arrow in a circle at the top left of the Arduino IDE) to upload your code to your Arduino and run your program

At this stage your code should begin running in the default mode: test your device is working!

Modifications and updates to the code

- Line 7 of the code specifies the minimum distance to an object before starting to beep:
  - `int const max_distance = 100;`
- Try modifying the value "100" here and re-uploading your code (using the upload button) to see if this makes it easier or hard to navigate
- Try commenting out line 54 of the code by placing two forward slashes in front of it:
  - `//digitalWrite(BUZZERPIN, HIGH); // Buzz`
- And replace this line with:
  - `beep_rate(distance); // Alternative way of beeping`

This uses an alternative form of the code that changes the rate of beeping based on the distance to a detected object.