Distance Gun

Instructor Guide

Level 2

1. **Download the UltraDistSensor Library:**

   Student Hint: Go to www.BuildARobotWorkshop.com/---------- to download the zip of the library. To use this library go to :

   Sketch → Include Library → Add .ZIP Library → (Click on the Downloaded Library)

   Instructor guide: This library will be used to communicate with the ultrasonic sensor. It takes care of pinging objects and calculating distance based on speed of sound for the user with two functions. These functions are `sensor.distanceInInch(); and sensor.distanceInCm();` Without a library to communicate with the Ultrasonic sensor, code cannot be written for it. Builders will download the library for the Arduino IDE and use it to find the distance to an object.

Notes:

_____

_____

2. **Explanation External Code Libraries**

   Instructor guide: Point out the functions `sensor.distanceInInch();and sensor.distanceInCm();` are used to find the distance between objects in different types of measurements.

   `#include <UltraDistSensor.h>` section on their coding sheet is an external library that allows the use of those functions. Explain the following:

   - External code libraries are commonly used in coding so that you can limit the amount of code you write.
   - These functions use sound to ping the object in front of them. Once the object reflects the sound waves they measure the distance using the amount of time it took for them to receive a signal back.
   - Some objects such as soft cloth absorb more sounds making it harder to measure the distance to those objects.
   - That is already written in the library and with the use of these functions we can detect distance.

3. **Complete Challenge 1:**

   Student Hint: To makes sure everything you printed appears on screen, change the X and Y coordinates of where you printed each word.

   Instructor guide: This challenge will teach the builder to use functions to find the readings from the Ultrasonic sensor. They will track the distance in inches, centimeters, and print those

distances on screen. Additionally they will also print the difference between both of those readings on the display to become more familiar with those functions and the lcd.print() function.

Notes:

_____

_____

CODE The following code is a sample solution for Challenge 1.

```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <UltraDistSensor.h>

UltraDistSensor scanner;
LiquidCrystal_I2C lcd(ADDRESS, 20, 4);

float inchReading;
float cmReading;

void setup() {

    lcd.init();
    lcd.backlight();
    scanner.attach(9, 10);

}

void loop() {

    lcd.scrollDisplayRight();

    inchReading = scanner.distanceInInch();
    cmReading = scanner.distanceInCm();

    lcd.setCursor(0, 0);
    lcd.print("Inches: ");
    lcd.print(inchReading);

    lcd.setCursor(11, 0);
    lcd.print("Cm: ");
    lcd.print(cmReading);
```

```
    lcd.setCursor(0, 1);
    lcd.print("Cm - Inches: ");
    lcd.print(cmReading - inchReading);
    delay(400);

}
```

Notes:

---

4. **Explanation If and Else Statements**

   Student Hint: Go to www.BuildARobotWorkshop.com/---------- and watch the provided video to better understand the following concept.

   Instructor guide: Explain that if and else statements are used to execute certain lines of code only if a particular statement is true. Else if statements can also be used to execute certain code if it meets a different condition instead.

   Explain the following:

   ```
   if(condition 1 is true){
       // Execute Code 1
   } else if(condition 2 is true){
       // Execute Code 2
   } else {
       // Execute Code 3
   }
   ```

   - The format above is an example of what the code will be executed if a certain condition is met.
   - Examples of a condition are `x > 1` or `y == 5`.

   IMPORTANT: IF A CONDITION IS **TRUE**, THE CODE WILL **EXECUTE** A CERTAIN PIECE OF CODE AND **EXIT** THE IF STATEMENT. IT WILL NO LONGER CHECK THE **ELSE IF** AND **ELSE** STATEMENTS.

Notes:

---

5. **Explanation Operators**

Student Hint: Go to www.BuildARobotWorkshop.com/---------- and watch the provided video to better understand the following concept.

Instructor guide: Point out the operators `<=, ==, >=, <, >, &&, !=, and ||` are used to write conditional statements. These operators compare 2 different measurements and can be used to execute certain pieces of code if a particular statement is true. They return true or false booleans depending on what the statements evaluate to. What they return can be used in if statements.

Explain the following:

- `x <= y` : This operator checks if the x (the left side) is less than or equal to y (the right side).
- `x >= y` : This operator checks if the x (the left side) is greater than or equal to y (the right side).
- `x < y` : This operator checks if the x (the left side) is less than y (the right side).
- `x > y` : This operator checks if the x (the left side) is greater than y (the right side).
- `x == y` : This operator checks if the x (the left side) equals to y (the right side).
    - `==` is different from `=` as the first one checks if two objects are equal to each other while the second one sets the first object equal to the second one
- `x != y` : This operator checks if the x (the left side) is not equal to y (the right side).
- `x && y` : This operator checks if the condition x (the left side) is true and the condition y (the right side) is true.
    - If both conditions are true then this statement returns true.
    - In this case `x and y` are both conditions themselves. For example they could be one of the previous lines.
- `x || y` : This operator checks if the condition x (the left side) is true or the condition y (the right side) is true.
    - If one condition is true then this statement returns true.
    - In this case `x and y` are both conditions themselves. For example they could be one of the previous lines.

Notes:

_____

_____

6. **Complete Challenge 2:**

   Student Hint: To gain the largest and smallest readings make sure to consider the operators used carefully.

   Instructor guide: This challenge will teach the builder to use conditional statements and operators to find specific readings from the Ultrasonic sensor. They will track the largest and smallest distance and store them in specific variables. They will then print these variables on the screen. This challenge is an opportunity for the builder to explore the range of their Ultrasonic LCD with just their current code.

Notes:

CODE The following code is a sample solution for Challenge 2.

```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <UltraDistSensor.h>

UltraDistSensor scanner;
LiquidCrystal_I2C lcd(ADDRESS, 20, 4);

float distance;
float largestDistance;
float smallestDistance;

void setup() {

    lcd.init();
    lcd.backlight();
    scanner.attach(9, 10);

}

void loop() {

    lcd.clear();
    distance = scanner.distanceInInch();
```

```
    if(distance > largestDistance){
        largestDistance = distance;
    }

    if(distance < smallestDistance) {
        smallestDistance = distance;
    }

    lcd.setCursor(0, 0);
    lcd.print("Greatest: ");
    lcd.print(largestDistance);

    lcd.setCursor(0, 1);
    lcd.print("Smallest: ");
    lcd.print(smallestDistance);

    delay(500);

}
```

Notes:

---

7. **Complete Challenge 3:**

   Student Hint: Make the comparisons of the messageNum variable and the newMessageNum variables are used correctly in if statements. (Use of the == and != operators).

   Instructor guide: This challenge will teach the builder to compare multiple variables to gain information about the distance and use that information to print special messages. There are 4 different messages to print based on the distance. The builder will store which messages it is (1 - 4) in the newMessageNum variable. Scrolling the text can only be done if the same message is going to be printed. To check, the old message and the new message numbers are compared. If those numbers are not the same then we change the text on the screen to the new message. Otherwise, we will scroll the text that was previously there.

Examples of Messages:

- If the distance is less than or equal to 20 inches print “Object is Very Close By”.
- If the distance is less than or equal to 40 inches and greater than 20 inches print “Object is not Too Close”.
- If the distance is less than 75 inches and greater than 40 inches print “Object is not Too far”.
- If the distance is greater than 75 inches print “Object is Far Away”.

Notes:

_____

_____

CODE The following code is a sample solution for Challenge 3.

```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <UltraDistSensor.h>

UltraDistSensor scanner;
LiquidCrystal_I2C lcd(ADDRESS, 20, 4);

float distance;
float messageNum = 0;
float newMessageNum = 0;

void setup() {

    lcd.init();
    lcd.backlight();
    scanner.attach(9, 10);

}

void loop() {

    distance = scanner.distanceInInch();
```

```
    if(distance <= 20){
      newMessageNum = 1;
    } else if(distance <= 40){
      newMessageNum = 2;

    } else if(distance <= 75){
      newMessageNum = 3;
    } else {
      newMessageNum = 4;
    }

    if(newMessageNum != messageNum){
        messageNum = newMessageNum;
        lcd.setCursor(0, 0);
        if(messageNum == 1){
             lcd.print("Object is Very Close By");
        } else if(message No == 2){
             lcd.print("Object is not Too Close");
        } else if(messageNum == 3){
             lcd.print("Object is not Too Far");
        } else {
             lcd.print("Object is Far Away");
        }
    } else {
        lcd.scrollDisplayLeft();
    }

    delay(200);

}
```

Notes:

_____

_____

POSSIBLE PROBLEMS AND ANSWERS

Problem: LCD display does not show text
Possible Solutions:

1. The index at which the text was printed at might leave enough room for the full message to be shown
2. The lcd.clear() function is making it unable to see any text being displayed.
3. The ground and VCC pins are not connected to the (-) and (+) on the breadboard.
4. The SDA pin is not connected to A4 and the SCL pin is not connected to A5 on the Arduino Uno.
5. No power is being supplied
6. LCD was not initialized in the code by using lcd.init() or lcd.backlight().

Problem: LCD does not scroll text
Possible Solutions:

1. The lcd is being cleared every loop (lcd.clear() is being used) so the text doesn't scroll every frame
2. The lcd may reset the text every time at a certain index. This action resets the text as the coordinate picked and does not allow for the text to be scrolled completely.