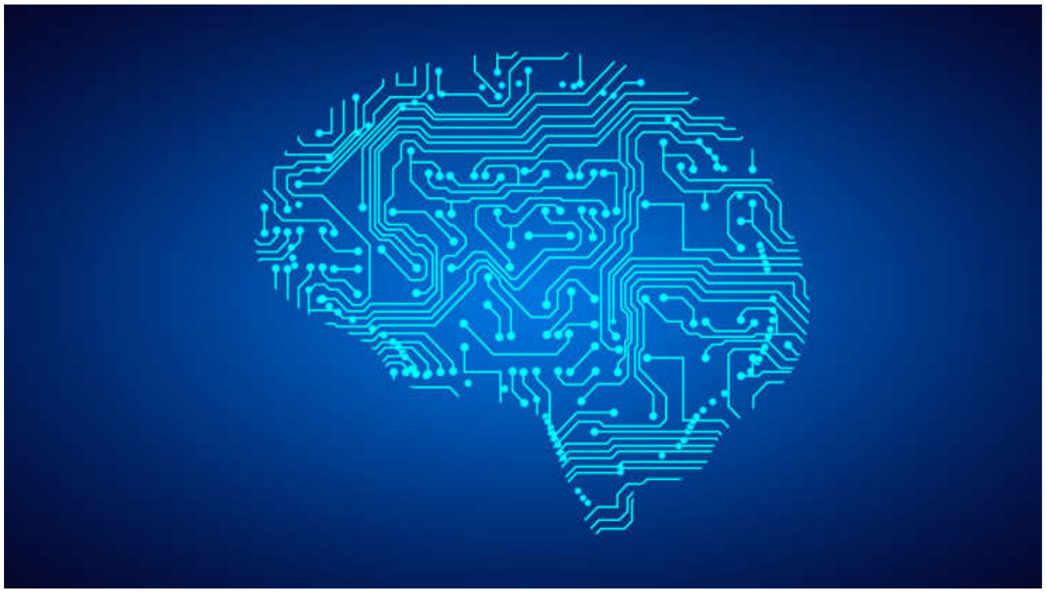
TRASH
ONLY

Group 70- Weekend Warriors

TRASH TALK

Your Friendly Neighborhood Smart Bin

01.12.2017



Project Preamble

Why we need Trash Talk-

In an overly and overtly exploitative era with an excess of everything for a certain fraction of the society, it is an obligation to be smart with waste disposal and management. We belong to a time when everything is controlled by technology, thus making our garbage disposal system "artificially intelligent" would be the smart thing to do. Many of us are not aware of the intricacies of waste segregation or basic waste disposal. Hence, we present to you Trash Talk- Your Friendly Neighborhood Smart Bin.

Project Scope

What Trash Talk does-

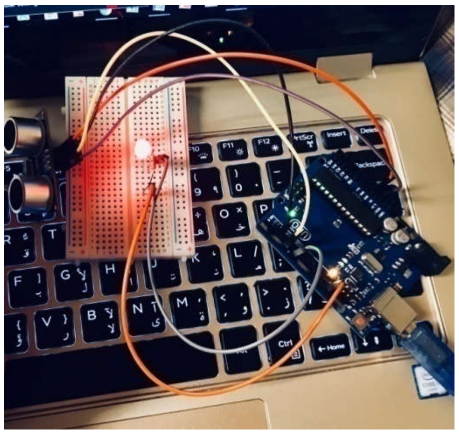
Trash Talk is an artificially intelligent waste management system that involves artificial intelligence, embedded-systems, logging, calculation, real-time sensing and data updates, and a communication system which enables Trash Talk to intelligently interact with the user, making him/her aware of the implications of proper waste management, presents a survey and logs the statistics from the survey. Trash Talk minimizes commute budget and saves time by sending out alerts to the concerned individual when the bin is full. It is, therefore, a *fully automated* Trash Management System.

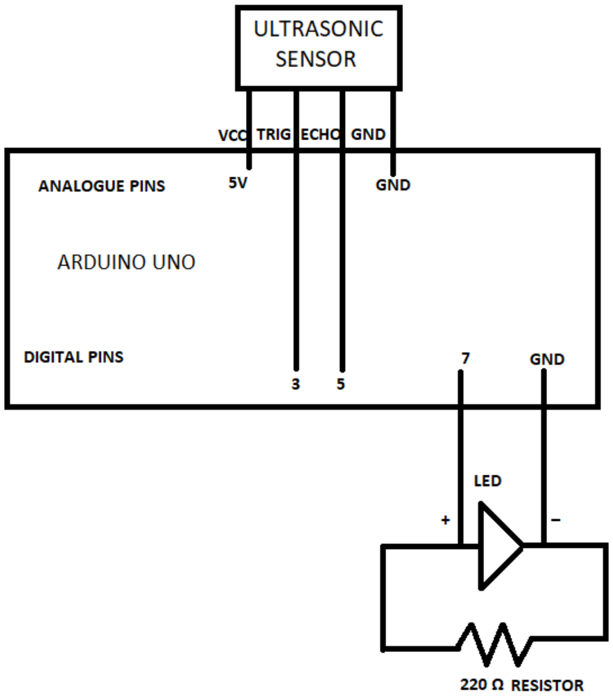
Systems Design

Components-

- Onion Omega2 and Power Dock
- Arduino Uno (Analogue to Digital Conversion)
- Breadboard, LED light bulbs, 220 ohm resistor, jumper wires
- Ping Ultrasonic Distance Sensor
- An amazing team!

Circuit Diagram-





REAL WORLD WIRING

CIRCUIT DIAGRAM

This part of the embedded-systems deals with sending our alerts if the trash bin is full. The Ping Ultrasonic Sensor has two components- a transmitter and a receiver. The 'trig' pin is for the output and the 'echo' pin is for the input. It gives us the time taken by the ultrasonic wave to travel from the transmitter to the receiver in microseconds. Our program outputs the distance on the serial monitor by multiplying the time with the speed of sound in air (both in inches/microsecond and cm/microsecond as per the user's convenience) which gives us the distance of the obstacle that reflects the wave from the sensor in inches and centimeters respectively. If the distance is less than or equal to 4 cm, our program outputs an alert message saying that the bin is full. By using a Third Party Software (eg: 1SHEELD) we can send the alert output that is being displayed on the serial monitor. The distance is constantly displayed on the serial monitor, thereby, computing real-time statistics of the fraction of the bin that is empty.

A second component to this part of the embedded systems is a red LED light with a 220 ohms resistor connected to it which blinks every second(every 1000 ms) when the trash bin is full to let the user (who has come to dispose of trash) know that the bin is full so that he/she uses a different bin.

The program to receive real-time data from the Ping Ultrasonic sensor, activate the LED and send the alert message is as follows:

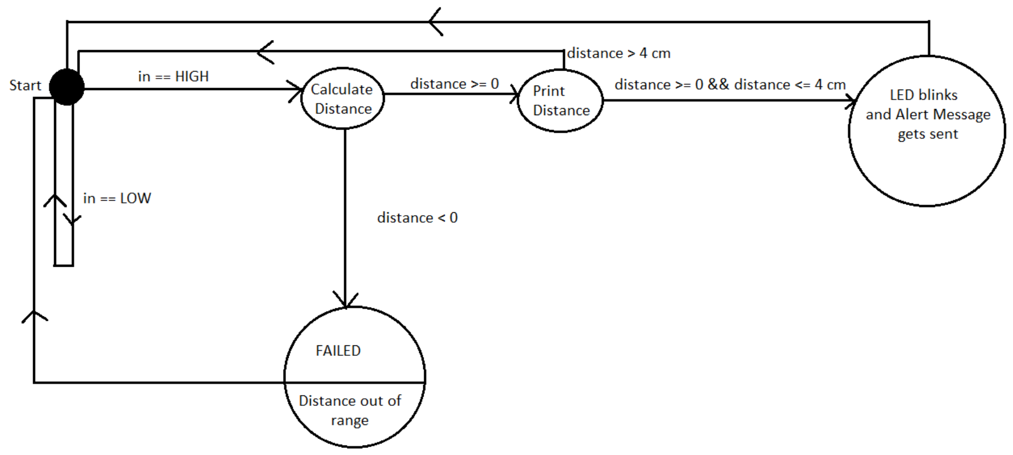
https://pastebin.com/vqUsdn3Q

The source-code of this component of the embedded-systems project has the following functionality:

void setup() //initializes serial communication

void loop() /* reads the echo pin: a HIGH pulse whose duration is the time (in microseconds) from the sending of the ping to the reception of the echo of its object and calculates the distance if the sensor is triggered by a HIGH pulse of 10 or more microseconds (it gives a short low pulse beforehand to ensure a clean HIGH pulse) by multiplying the duration with the speed of sound in air(340 m/s or 29 microseconds per centimeter). If the distance is less than or equal to 4 cm, it activates the LED which blinks every 1 second because of digitalWrite(LED,HIGH); delay(1000); digitalWrite(LED,LOW); Since 1000 milliseconds equal 1 second for the user to notice who comes to dispose of waste. It sends out an e-mail alert to the concerned authority (could be done using a third party software) to ensure removal of waste from the bin. The distance is constantly displayed on the serial monitor, thereby, computing real-time statistics of the fraction of the bin that is empty. */

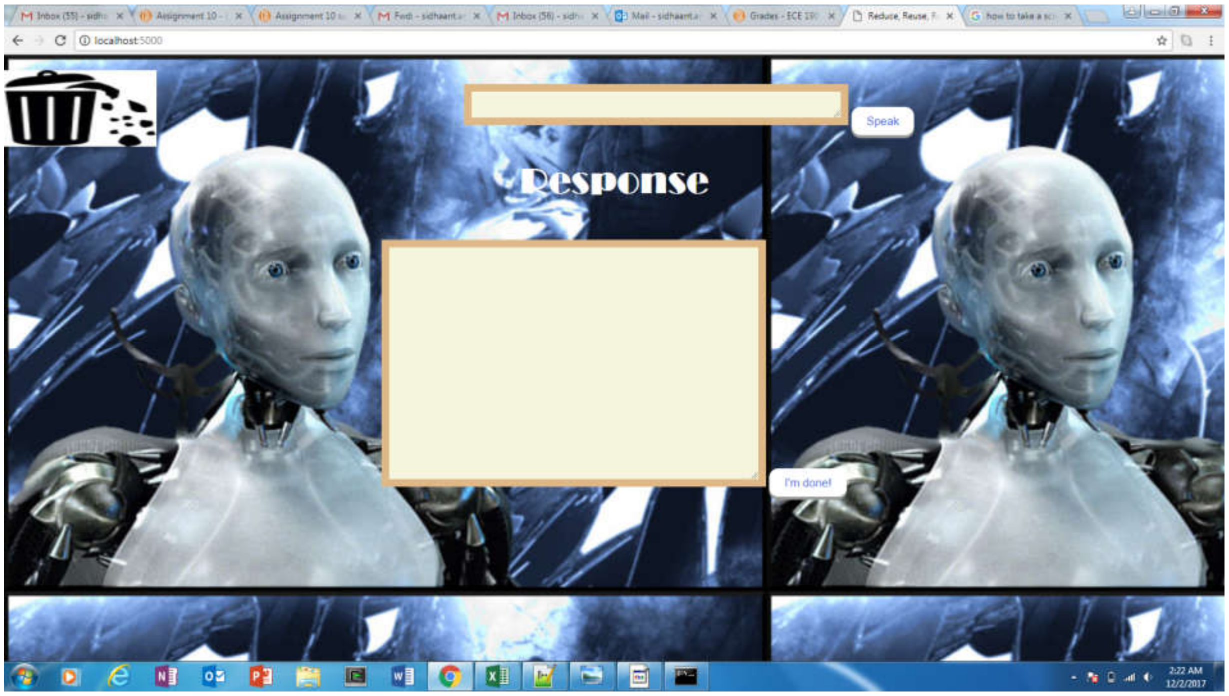
The State Machine Diagram of the Source Code:



Artificial Intelligence-

Besides the mandatory features of writing a code which includes the logging and statistics functionality of the embedded-systems project using an Omega-2, we added a *bonus* component to our project- Artificial Intelligence using Node.js. We used JavaScript to build an artificially intelligent chatbot which creates an interactive experience by guiding the user through the process of proper waste disposal. We created two separate Entities namely, Non-biodegradable and Biodegradable that have a huge database of the aforementioned types of wastes with a common pseudonym for each type. Whenever the user asks where he/she should dispose of a certain piece waste material, our program's Intent captures the said member of the Entity and outputs the stored response enabling Trash Talk to guide him/her accordingly. However, if the said object is not present in the chatbot's Entity, it intelligently asks the user what substance it is made of and determines the category of waste through a guided experience. To make the process more user-friendly and hassle free, we included the speech-to-text feature which converts the user's speech to text for our chatbot, and the text-to-speech feature which, in turn, converts our chatbot's text to speech for the user, thus eradicating the strenuous task of typing while disposing of waste. We have used SpeechRecognition and SpeechSynethesisUtterance on JavaScript for this task and given our chatbot the personality of "Agnes". We have used Cascading

Style Sheets to build the look and feel of our web page. We have also built a server to host our web page locally. Here's a glimpse of Trash Talk:



We are currently running the web page on our laptops, but we expect to have a built-in screen in front of every trash bin to maximize user experience.

The source code of our chat page is as follows:

https://pastebin.com/QBqVd2Pt

The <style> tag has a .button class for the look and feel of the buttons; it also designs the body of the chat page including the background image, text area design, text alignment and margins.

The <body> tag contains the logo image, textarea and uses api.ai to access our AI. The $(document).ready(function() is a jQuery that sends the user input if(event.which == 13), i.e., when the user presses enter. The function function startRecognition() contains the Web Speech API which converts speech to text using SpeechRecognition and text to speech using SpeechSynthesisUtterance (inside the function speak(string)). The function function send(text) sends the text to api.ai.

Extremely Simplified State Machine Diagrams of the Source Code stating the functionality of Safe Waste Disposal (Figure 1) and Speech-to-Text and Text-to-Speech Conversion (Figure 2) respectively

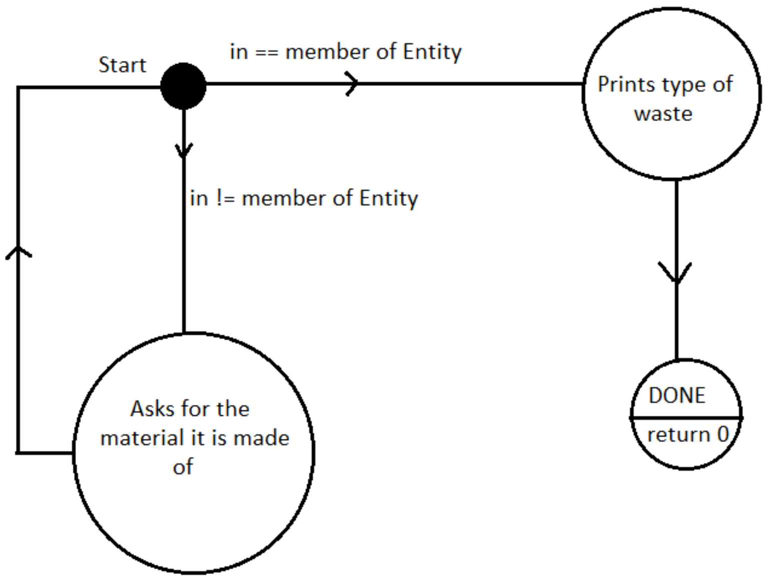


Figure 1

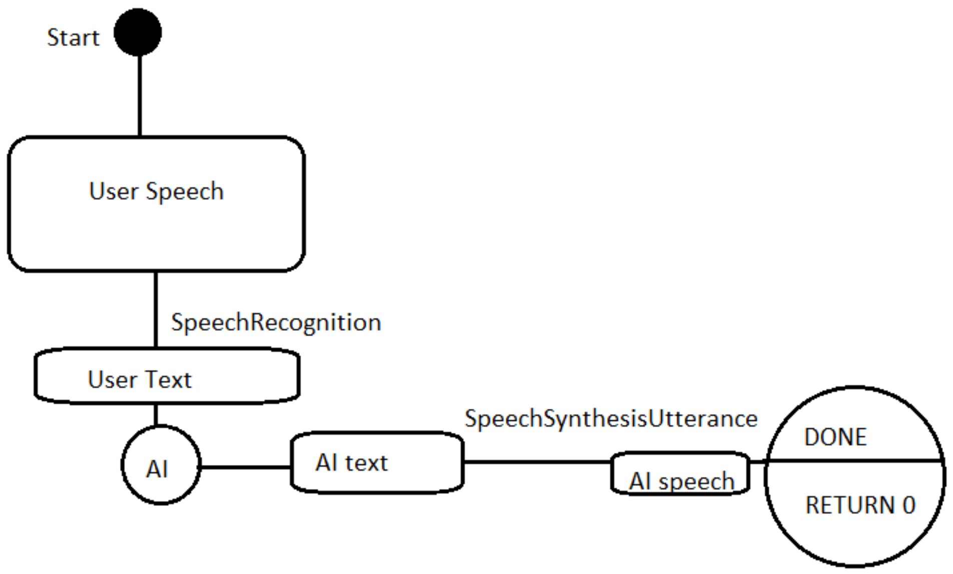


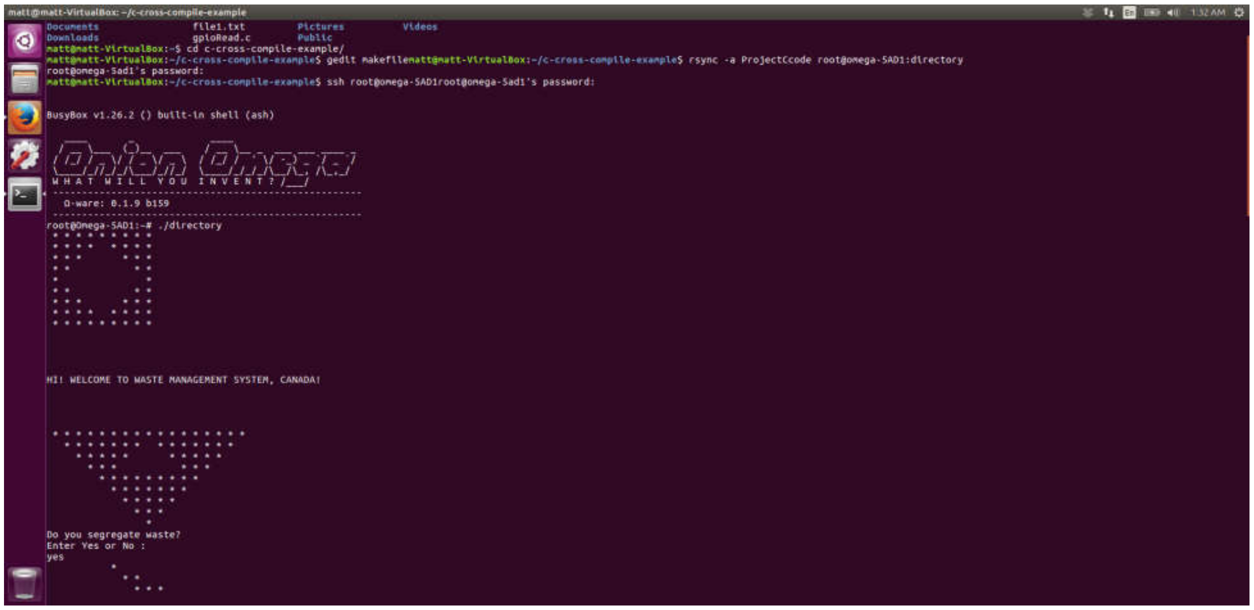
Figure 2

An Interactive Environmental Survey to determine Statistics

On the same screen that we expect to be installed in front of every smart bin, there shall be an interactive environmental survey at the end of waste disposal to gather important statistics and spread awareness.

We have implemented this using the Onion Omega-2 and the power dock. We have written a program in C language on the Ubuntu Virtual Box which *reads* a series of questions from a .txt file which the user has to answer (the user has two options- Yes or No). It stores the number of Yes's and No's in the self-same .txt file by *writing* to it. The program also performs statistical operations on the available data and displays the data to the user. We have compiled the said program on xCompile and uploaded it to the Omega2 along with the text file. In this segment of the embedded systems project, we have successfully utilized the skills of C language coding, Statistics Calculation and Data Logging (by writing to a file and reading form it) that we have developed over the length of the ECE-150 course.

Onion Omega2 with the Power Dock- an affordable, expandable and easy to use IoT Computer



```
matt@matt-VirtualBox: ~/c-cross-compile-example
Documents               file1.txt         Pictures          Videos
Downloads               gpioRead.c        Public
matt@matt-VirtualBox:~$ cd c-cross-compile-example/
matt@matt-VirtualBox:~/c-cross-compile-example$ gedit makefilematt@matt-VirtualBox:~/c-cross-compile-example$ rsync -a ProjectCcode root@omega-5AD1:directory
root@omega-5ad1's password:
matt@matt-VirtualBox:~/c-cross-compile-example$ ssh root@omega-5AD1root@omega-5ad1's password:


BusyBox v1.26.2 () built-in shell (ash)

 W H A T  W I L L  Y O U  I N V E N T ? /___/
 -----------------------------------------------------
   Ω-ware: 0.1.9 b159
 -----------------------------------------------------
root@Omega-5AD1:~# ./directory
 * * * * * * * * *
 * * * *   * * * *
 * * *       * * *
 * *           * *
 *               *
 * *           * *
 * * *       * * *
 * * * *   * * * *
 * * * * * * * * *



HI! WELCOME TO WASTE MANAGEMENT SYSTEM, CANADA!



 * * * * * * * * * * * * * * * * *
   * * * * * * *   * * * * * * *
     * * * * *       * * * * *
       * * *           * * *
         * * * * * * * * *
           * * * * * * *
             * * * * *
               * * *
                 *
Do you segregate waste?
Enter Yes or No :
yes
           *
             * *
               * * *
```

A glimpse of the Interactive Environmental Survey to spread awareness and compute Statistics over the received data by logging

This segment of the embedded-systems project uses the following source-code files:

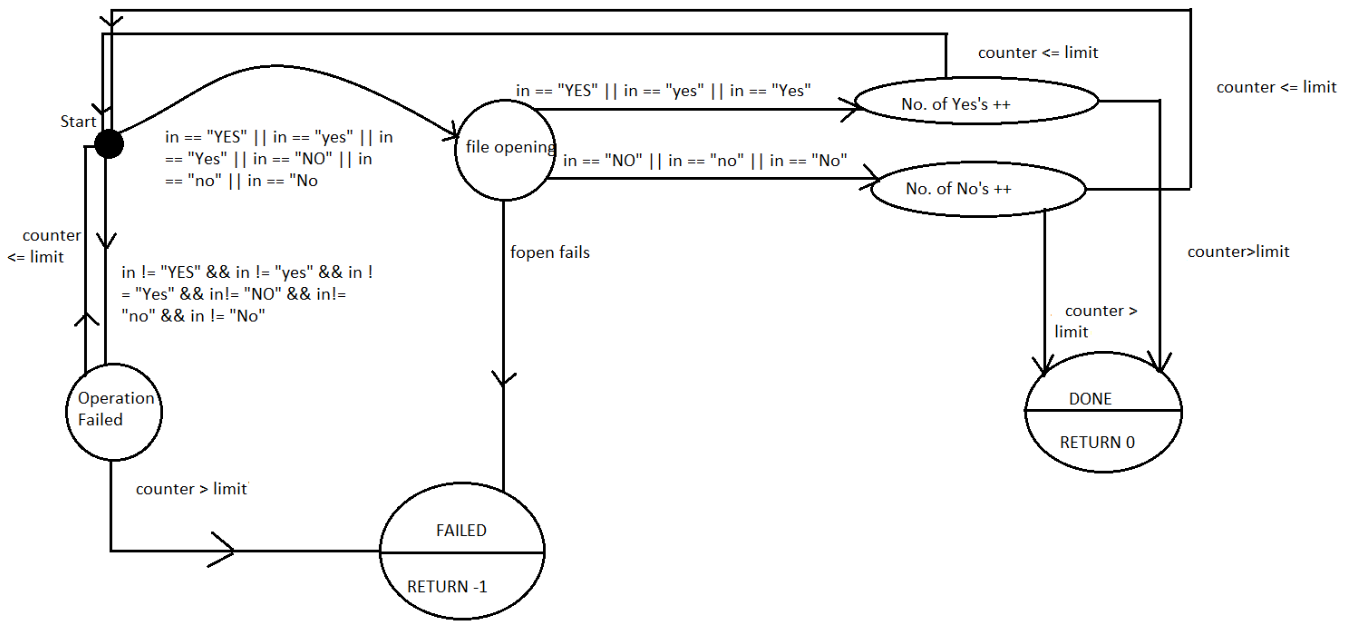
- ProjectCcode.c (https://pastebin.com/AEAn8sQp)
- cfile.txt (https://pastebin.com/kxqSFmCM)
- makefile (https://pastebin.com/ZBwvxfUe)

We are using xCompile.sh to compile our program.

The ProjectCcode.c source-code file has the following functions:

- void initializeQues() //copies the questions into the ques[] array
- void initializeAns() //initializes the answers
- void initializeOptions() //initializes the options
- int askQuestion(char *currQuestion) //asks a question(function is called using a 'for' loop) and stores the answer by writing to the file
- int printStatistic() //reads the values from the file and computes statistics
- void patternintro() //prints the introduction pattern when called
- void pattern0() //prints the pattern for even-numbered questions (starting from 0)
- voidpattern1() //prints the pattern for odd-numbered questions
- int triangle (int nos, int i, int skip) //helper function called by patternintro()
- int main() //this is the main function running the for loop

The State Machine Diagram of the Source Code:


Start
in == "YES" || in == "yes" || in == "Yes" || in == "NO" || in == "no" || in == "No
file opening
in == "YES" || in == "yes" || in == "Yes"
in == "NO" || in == "no" || in == "No"
No. of Yes's ++
No. of No's ++
counter <= limit
counter <= limit
counter>limit
counter > limit
DONE
RETURN 0
fopen fails
counter <= limit
in != "YES" && in != "yes" && in != "Yes" && in!= "NO" && in!= "no" && in != "No"
Operation Failed
counter > limit
FAILED
RETURN -1

Subset achieved

What Trash Talk does in real life-

As mentioned in the project proposal, Trash Talk creates a guided user experience for safe waste disposal and sends out alerts both by lighting up the red LED and intimating the authorities when the Ping Ultrasonic Distance Sensor senses that the bin is full. We initially planned to create a daily/weekly/monthly/annual waste report. However, we replaced that with a waste disposal survey that takes input from the user in real-time and computes statistics by reading from and writing to a file.

Ways to improve the project

What we would like to add to Trash Talk in the future-

We intend on activating a noise alarm if a user drops waste outside the bin (within a 1 m radius) thereby ensuring proper waste disposal. The challenge here is to distinguish between the user’s presence and the presence of a garbage bag. The alarm should only go off if the waste stays on the floor a sufficient period of time.

The obstacle we encountered while working with the Omega2 was the fact that it could not process analogue signals which is why we had to resort to the Arduino Uno for the conversion of analogue signals from the Ping Ultrasonic Sensor to digital signals. In the future we would like to integrate the Arduino and the Omega2 by ***interfacing the Onion Omega2 and the Arduino Uno via UART (details provided in** Appendix G**).***

Things learnt from the project

(Besides the fact that Section 001 is cooler)

This term's final project taught us what a team activity is in the true sense of the word. From brainstorming through implementation to testing, everything was collaborative and a learning experience for every team member present in the room. We learnt to put our differences aside and realized that it is okay to agree to disagree sometimes as long as it does not affect the joint venture. We learnt that setting the system up takes more time than writing the actual source code. Murphy's law reinforced its existence- "Anything that can go wrong, will go wrong." The decision of keeping buffer time on our hands was, therefore, a very judicious decision on our part. Working on the project gave us a stand-alone approach to solving a problem while also working in a group without the supervision of an instructor and exposed us to electronics and their functionality to an extent beyond the usual classroom-learning. We realized the delicate relationship between software programming and hardware compatibility and how they complement each other. As was required of us, we now have an in-depth understanding of designing and developing an embedded-systems project, cross-compilation, the use of logging for embedded-systems debugging, statistics computation and the use of system-layer abstraction for embedded-system design.

Had we a chance to redo it

What we would do differently-

If we could have a do-over, we would scrutinize the hardware details ahead of time and sketch out a more detailed time-frame with an increased number of short term goals. Bust most importantly, we would buy more jumper wires! We would also like to divide the responsibilities equally among all the team-members, allocating each one enough time to acquire expertise in his/her respective field. We would try and have fewer fights over implementation!

Pro-tip

Advice to someone who would be working on a similar project-

Our advice to anyone starting this project is to start early and work on it piece by piece. Working on it incrementally will allow anyone to see simple mistakes in data path and instruction set early on. That being said, we would advise future students to spend a lot of time on the data path and instruction set to prevent headaches and last minute fixes being applied toward the end of the project when it is more difficult to add extra functionality. Another piece of advice would be to not be afraid of asking questions to the TA's. David and Matt in our case were more than helpful in explaining server issues and possible solutions to library problems that would have taken a long time on our own to come up with. In conclusion, our advice would be not to stress over it. It is a fun project in a class that is rather complicated. If things do not make sense the first time you encounter them, take a step back and ask for help or explanation if needed. Do not waste time asking your classmates set-up questions; keep trying until you succeed; they know as less as you. The internet has a hoard of websites a few of which shall be of use to you. Don't forget to have fun, learn a lot and ask yourself why you are doing what you are doing!

Appendices

Appendix A- Artificial Intelligence and Machine Learning: The most important general-purpose technology of our era is artificial intelligence, particularly machine learning (ML) — that is, the machine's ability to keep improving its performance without humans having to explain exactly how to accomplish all the tasks it's given. Within just the past few years machine learning has become far more effective and widely available. We can now build systems that learn how to perform tasks on their own.

https://hbr.org/cover-story/2017/07/the-business-of-artificial-intelligence

Appendix B- api.ai: Uses machine learning to *understand* what users are saying using years of domain knowledge and natural language understanding, api.ai analyzes and understands the user's intent and responds in the most useful way.

https://dialogflow.com/docs/getting-started/basics

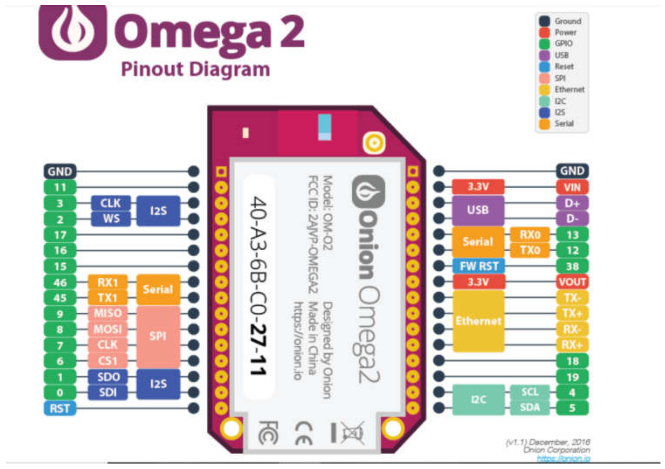
Appendix C- 1SHEELD: A Third Party Software that turns one's Smartphone into 40 different Arduino shields (this was unfortunately out of our budget). This can be used to send e-mails to the authorities once the bin is full.

https://1sheeld.com/

Appendix D- Web Speech API: The Web Speech API enables one to incorporate voice data into web apps. The Web Speech API has two parts: SpeechSynthesis (Text-to-Speech), and SpeechRecognition (Asynchronous Speech Recognition).

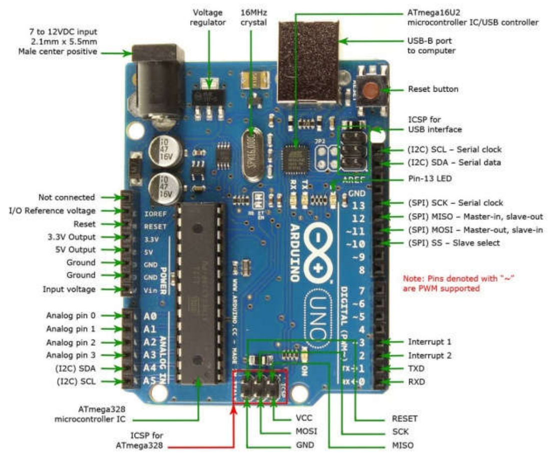
https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API

Appendix E- Onion Omega-2 and Power Dock: Omega2- an easy-to-use, expandable and affordable IoT computer. It is a Linux computer designed specifically for building connected hardware applications. It combines the tiny form factor and power efficiency of the Arduino, with the power and flexibilities of the Raspberry Pi. The Omega2 is highly integrated, has great IO, is Arduino compatible and modular and has internet connection. It is comes in two versions and is a full Linux computer!



https://onion.io/omega2/

Appendix F- Arduino Uno: We have used an Arduino Uno for converting the analogue signals of our Ping Ultrasound Distance Sensor to digital signals. Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.



https://store.arduino.cc/usa/arduino-uno-rev3

Appendix F: UART or Universal Asynchronous Receiver/Transmitter is a form of 1-to-1 serial communication, which relies on three connections between the two devices. One to transmit data, one to receive data and one for common ground. The below diagram demonstrates this. However in the case of the Arduino Uno and Omega2, we will be using a simple voltage divider circuit as well. This is just to avoid damaging the Omega2's internal circuitry, as the Omega2 cannot tolerate anything above 3.3v as an input. To test the serial communication, we will be using the 'screen' command on the Omega2 side and the 'serial monitor' tool on the Arduino IDE.

https://www.hackster.io/supersonnic/interfacing-the-onion-omega2-and-arduino-uno-via-uart-3f953d

Source Code:

The Interactive Environmental Survey Code in C Language(ProjectCcode.c):

https://pastebin.com/AEAn8sQp

```c
#include <stdio.h>
#include <time.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
```

```

char ques[7][100];
char options[5][100];
char ans[2][4];


void initializeQues()
{
    strcpy(ques[0], "Do you segregate waste?");
    strcpy(ques[1], "Did you trash non-biodegradable trash today?");
    strcpy(ques[2], "Did you waste food today?");
    strcpy(ques[3], "Do you know how many kids die of starvation in Africa?");
    strcpy(ques[4], "Do you see people throwing food around you everyday?");
    strcpy(ques[5], "Do you do your bit?");
    strcpy(ques[6], "Are you happy with your life?");

}

void initializeAns()
{
    strcpy(options[0], "Non-Woven Fabric");
    strcpy(options[0], "Ribbon");
    strcpy(options[0], "Paper Cups");
    strcpy(options[0], "Nylon");
    strcpy(options[0], "Thermocol");

}

void initializeOptions()
{
    strcpy(options[0], "Yes");
    strcpy(options[0], "No");
}

int askQuestion(char *currQuestion)
{
    // printf("LOL");
//  printf(currQuestion);

  puts(currQuestion);
  puts( "Enter Yes or No :");
   char currAnswer[4];
   gets(currAnswer);

   if((strcmp(currAnswer,"yes") == 0) || (strcmp(currAnswer,"YES") == 0))
   {
       strcpy(currAnswer,"Yes");

   }
   if((strcmp(currAnswer,"no") == 0) || (strcmp(currAnswer, "NO") == 0))
        strcpy(currAnswer,"No");
   if((strcmp(currAnswer,"Yes") != 0) && (strcmp(currAnswer,"No") != 0))
   {
       puts("Don't be a rebel, please put in the correct value!");
       return 0;
   }


//  puts(currAnswer);

    /* if(currAnswer != options[0] || currAnswer != options[1])
    {
        printf("IN1");
        return -1;
    }
else
{ */

        FILE *ifp;
        char *mode = "r";
        int c;

        ifp = fopen("cfile.txt", mode);

         if (ifp == NULL) {
         fprintf(stderr, "Can't open input file in.list!\n");
         printf("cant open file");
         exit(1);
         return -1;
        }



    /*  char readQues[7][100];
        char yeses[7];
        char nos[7];

        for(int line = 0; line<7; line++)
        {

         yeses[line] = getc(ifp);
        printf("%c", yeses[line]);
         getc(ifp);
         nos[line] = getc(ifp);
         printf("%c", nos[line]);
         getc(ifp);

         char toPut[100];
         int a;
         int k = 0;
        while('\n' != (a = fgetc(ifp))) {
            toPut[k] = a;
            k++;
        printf("%c",a);
    }



         strcpy(readQues[line], toPut);
        // printf(readQues[line]);
        }
 //  } */


 // extract intoo lines
 char lines[7][1000];


    char ch = getc ( ifp );
    int line = 0;
    int index = 0;

    while ( ch != EOF ) {

        if ( ch != '\n'){

            lines[line][index++] = ch;
        }else {

             lines[line][index] = '\0';

            // printf("%i",line); printf(lines[0]);
            index = 0;
            line++;

            // printf (lines[line] );
        }
        ch = getc ( ifp );

    }

/*  for(int i = 0; i<7; i++)
```

```
        puts(lines[i]); */


/*
        char readQues[7][100];
        char yeses[7];
        char nos[7];
        int yesnocount = 0;
        int readQuesCount = 0;

        for(int line = 0; line <7; line++)
        {
            char c = lines[line][0];
            puts("IN");
            int  k = 0;
            while(c != '\0')
            {

                printf("%",c);
                if(k==0)
                {
                    yeses[yesnocount] = lines[line][0];
                }

                else if(k==2)
                {
                    nos[yesnocount] = lines[line][2];
                    yesnocount++;
                }

                else
                {
                    readQues[line][readQuesCount] = lines[line][k];
                    readQuesCount++;
                }

                k++;
            }
        }
        */

        // extract int yes, no and ques
        char readQues[7][100];
        char yeses[7][3];
        char nos[7][3];
        int yindex = 0;
        int nindex = 0;
        int rqindex = 0;
        int k = 0;

        for(int line = 0; line <7; line++)
        {
            k = 0;
            yeses[line][0] = lines[line][0];
            yeses[line][1] = lines[line][1];
            yeses[line][2] = '\0';
            // puts(yeses[line]);
            nos[line][0] = lines[line][2];
            nos[line][1] = lines[line][3];
            nos[line][2] = '\0';

            int currindex = 4;
            char curr = lines[line][4];
            while(curr != '.')
            {
                //printf("%c",curr);
                readQues[line][k] = curr;
                k++;
                currindex++;
                curr = lines[line][currindex];
            }

            readQues[line][k] = '\0';


        }

    /*  for(int i = 0; i<7; i++)
        puts(readQues[i]); */

    fclose(ifp);


    int found = 0;

    // puts(currQuestion);
    for(int i = 0; i<7;i++)
    {
        if(strcmp(currQuestion,readQues[i]) == 0)
            found = i;

    }

    FILE *ofp = fopen("cfile.txt", "w");


    for(int i = 0; i<7; i++)
    {
        if(i == found)
        {
            if(strcmp(currAnswer,"Yes") == 0)
            {
                char no_of_yes = yeses[i][0] - '0';
                no_of_yes = no_of_yes + 1;
                char putitback = no_of_yes  + '0';
                // printf("%c",putitback);
                yeses[i][0] = putitback;

            fputs(yeses[i], ofp);
            fputs(nos[i],ofp);
            fputs(readQues[i], ofp);
            fputc('.',ofp);
            fputc('\n',ofp);

        }

        if(strcmp(currAnswer,"No") == 0)
            {
                char no_of_no = nos[i][0] - '0';
                no_of_no = no_of_no + 1;
                char putitback = no_of_no + '0';
            //  printf("%c",putitback);
                nos[i][0] = putitback;

            fputs(yeses[i], ofp);
            fputs(nos[i],ofp);
            fputs(readQues[i], ofp);
            fputc('.',ofp);
            fputc('\n',ofp);

        }


        }
        else
        {
            fputs(yeses[i], ofp);
            fputs(nos[i],ofp);
            fputs(readQues[i], ofp);
            fputc('.',ofp);
            fputc('\n',ofp);
        }
    }

    fclose(ofp);


return 0;

}



int printStatistic()
{
    FILE *ifp;
        char *mode = "r";
        int c;

        ifp = fopen("cfile.txt", mode);

         if (ifp == NULL) {
         fprintf(stderr, "Can't open input file in.list!\n");
         printf("cant open file");
         exit(1);
         return -1;
        }

        char lines[7][1000];


    char ch = getc ( ifp );
    int line = 0;
    int index = 0;

    while ( ch != EOF ) {

        if ( ch != '\n'){

            lines[line][index++] = ch;
        }else {

             lines[line][index] = '\0';

            // printf("%i",line); printf(lines[0]);
            index = 0;
            line++;

            // printf (lines[line] );
        }
        ch = getc ( ifp );

    }

    char readQues[7][100];
        char yeses[7][3];
        char nos[7][3];
        int yindex = 0;
        int nindex = 0;
        int rqindex = 0;
        int k = 0;

        for(int line = 0; line <7; line++)
        {
            k = 0;
            yeses[line][0] = lines[line][0];
            yeses[line][1] = lines[line][1];
            yeses[line][2] = '\0';
            // puts(yeses[line]);
            nos[line][0] = lines[line][2];
            nos[line][1] = lines[line][3];
            nos[line][2] = '\0';

            int currindex = 4;
            char curr = lines[line][4];
            while(curr != '.')
            {
                //printf("%c",curr);
                readQues[line][k] = curr;
                k++;
                currindex++;
                curr = lines[line][currindex];
            }

            readQues[line][k] = '\0';


        }

    /*  for(int i = 0; i<7; i++)
        puts(readQues[i]); */

    fclose(ifp);

    for(int line = 0; line <7; line++)
    {
        puts(readQues[line]);
        puts("Number of Yeses :");
        puts(yeses[line]);
        puts("Number of Nos :");
        puts(nos[line]);
    }

    // calculating mean
        int sum1 = 0;
    int sum2 = 0;
    for(int line = 0; line<7;line++)
    {
        sum1 = sum1 + (yeses[line][0] - '0');
        sum2 = sum2 + (nos[line][0] - '0');
    }

    float meany = (float)sum1/7;
    float meann = (float)sum2/7;

    puts("Mean of the number of yeses : ");
    printf("%f\n", meany);
    puts("Mean of the number of nos : ");
    printf("%f\n", meann);

    int numYes[7];
    int numNo[7];

    for(int line = 0; line<7;line++)
    {
        numYes[line] = yeses[line][0] - '0';
        numNo[line] = nos[line][0] - '0';
    }

    // sorting
    int tmp = 0;
    for(int i = 0; i<7; i++)
    {
        for(int j = 0; j<7 - i - 1; j++)
        {
            if(numYes[j] > numYes[j+1])
            {
                tmp = numYes[j];
                numYes[j] = numYes[j+1];
                numYes[j+1] = tmp;
            }
        }
    }
```

```

    tmp = 0;
    for(int i = 0; i<7; i++)
    {
        for(int j = 0; j<7 - i - 1; j++)
        {
            if(numNo[j] > numNo[j+1])
            {
                tmp = numNo[j];
                numNo[j] = numNo[j+1];
                numNo[j+1] = tmp;
            }
        }
    }

    puts("Median of the number of yeses :");
    printf("%i\n", (numYes[3]));

    puts("Median of the number of nos :");
    printf("%i\n", (numNo[3]));

    // standard sampleDeviationn

    float sum3 = 0;
    for (int i = 0; i < 7; i++)
    sum3 = sum3 + (((float)numYes[i]) - ((float)(sum1/7))) * (((float)numYes[i]) - ((float)(sum1/7)));

    float sampleDeviation1 = sqrt((sum3 / (6)));

    puts("Standard Sample Deviation of  the number of yeses");
    printf("%f\n", (sampleDeviation1));

    float sum4 = 0;
    for (int i = 0; i < 7; i++)
    sum4 = sum4 + (((float)numNo[i]) - ((float)(sum2/7))) * (((float)numNo[i]) - ((float)(sum2/7)));

    float sampleDeviation2 = sqrt((sum4 / (6)));

    puts("Standard Sample Deviation of  the number of nos");
    printf("%f\n", (sampleDeviation2));

    //  population sample deviation

    float popDeviation1 = sqrt((sum3 / (7)));
    float popDeviation2 = sqrt((sum4 / (7)));

    puts("Population Sample Deviation of  the number of yeses");
    printf("%f\n", (popDeviation1));
    puts("Population Sample Deviation of  the number of nos");
    printf("%f\n", (popDeviation2));



    // Mode



}

void pattern0()
{
 char prnt = '*';
 int i, j, k, s, sp, nos = 0, nosp = -1;
 for (i = 9; i >= 3; (i = i - 2)) {
  for (s = nos; s >= 1; s--) {
   printf("  ");
  }
  for (j = 1; j <= i; j++) {
printf("%2c", prnt);
}
for (sp = nosp; sp >= 1; sp--) {
   printf("  ");
  }
  for (k = 1; k <= i; k++) {
 if (i == 9 && k == 1) {
continue;
}
printf("%2c", prnt);
}
nos++;
nosp = nosp + 2;
printf("\n");
}
nos = 4;
for (i = 9; i >= 1; (i = i - 2)) {
  for (s = nos; s >= 1; s--) {
   printf("  ");
  }
  for (j = 1; j <= i; j++) {
   printf("%2c", prnt);
  }
  nos++;
  printf("\n");
 }
}
int triangle(int nos, int i) {
 char prnt = '*';
 int s, j;
 for (s = nos; s >= 1; s--) {    // Spacing factor
  printf("  ");
 }
 for (j = 1; j <= i; j++) {      //The inner loop
  printf("%2c", prnt);
 }
 return 0;
}

void pattern1() {
 int i, nos = 5;
 //draws the upper triangle
 for (i = 1; i <= 4; i++) {
 triangle(nos, i);    //Inner loop construction
 nos++;              // Increments the spacing factor
 printf("\n");  }
nos = 7;  //Draws the lower triangle skipping its base.
for (i = 3; i >= 1; i--) {
  int j = 1;
  triangle(nos, i);  // Inner loop construction
  nos = nos - j;     // Spacing factor
  printf("\n");
 }
}
int triangle1(int nos, int i, int skip) {
 char prnt = '*';
 int s, j;
 for (s = nos; s >= 1; s--) {
  printf("  ");
 }
 for (j = 1; j <= i; j++) {
 if (skip != 0) {
 if (i == 9 && j == 1) {
   continue;
 }
 }
if (i == 1 || i == 9) {
 printf("%2c", prnt);
}
else if (j == 1 || j == i) {
 printf("%2c", prnt);
 } else {
 printf("  ");
}  }
```

```
return 0; }
void pattern() {
int i, nos = 0, nosp = -1, nbsp = -1;
for (i = 9; i >= 1; (i = i - 2)) {
  triangle1(nos, i, 0);
  triangle1(nosp, i, 1);
  triangle1(nbsp, i, 1);
  printf("\n");
  nos++;
  nosp = nosp + 2;
  nbsp = nbsp + 2;
 }
 nos = 3, nosp = 5, nbsp = 5;
 for (i = 3; i <= 9; (i = i + 2)) {
  triangle1(nos, i, 0);
  triangle1(nosp, i, 1);
  triangle1(nbsp, i, 1);
  printf("\n");
  nos--;
  nosp = nosp - 2;
  nbsp = nbsp - 2;
 }
}
void patternintro() {
 char prnt = '*';
 int i, j, k, s, nos = -1;
 for (i = 5; i >= 1; i--) {
  for (j = 1; j <= i; j++) {
 printf("%2c", prnt);
}
for (s = nos; s >= 1; s--) {
    printf("  ");
   }
   for (k = 1; k <= i; k++) {
    if (i == 5 && k == 5) {
     continue;
    }
    printf("%2c", prnt);
   }
   nos = nos + 2;
   printf("\n");
  }
 nos = 5;
 for (i = 2; i <= 5; i++) {
  for (j = 1; j <= i; j++) {
 printf("%2c", prnt);
}
for (s = nos; s >= 1; s--) {
    printf("  ");
   }
   for (k = 1; k <= i; k++) {
    if (i == 5 && k == 5) {
     break;
    }
    printf("%2c", prnt);
   }
   nos = nos - 2;
   printf("\n");
  }
 }
int main()
{
    initializeQues();
    initializeAns();
    initializeOptions();
    patternintro();
    puts(" ");
    puts(" ");
    puts(" ");
    puts(" ");
    puts("HI! WELCOME TO WASTE MANAGEMENT SYSTEM, CANADA!");
    puts(" ");
    puts(" ");
    puts(" ");
    puts(" ");
    for(int i = 0; i<7;i++) {
        if(i%2 == 0)
            pattern0();
        else
            pattern1();
    askQuestion(ques[i]);
    }

    puts("Do you want me to display the stastics");
    char input[4];
    gets(input);
    if((strcmp(input,"Yes") == 0) || (strcmp(input,"yes") == 0) || (strcmp(input,"YES") == 0))
        printStatistic();
    pattern();
    puts("THANK YOU FOR YOUR TIME! HAVE A TRASHFREE DAY!");
}
```

Arduino code in C Language(sensor.ino):

https://pastebin.com/vqUsdn3Q

```
const int trigPin = 5;

const int echoPin = 3;

const int LED = 7;


void setup() {

  // initialize serial communication:

  Serial.begin(9600);

  pinMode(LED, OUTPUT);

}


void loop()

{

  // establish variables for duration of the ping,
  // and the distance result in inches and centimeters:
  long duration, inches, cm;

  // The sensor is triggered by a HIGH pulse of 10 or more microseconds.
  // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
  pinMode(trigPin, OUTPUT);
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Read the signal from the sensor: a HIGH pulse whose
  // duration is the time (in microseconds) from the sending
  // of the ping to the reception of its echo off of an object.
  pinMode(echoPin, INPUT);
  duration = pulseIn(echoPin, HIGH);

  // convert the time into a distance
  inches = microsecondsToInches(duration);
  cm = microsecondsToCentimeters(duration);
  if(cm <0) {
   Serial.print("Out of range!");
  }
```

```
  else if(cm >= 0 && cm <= 4) {
   digitalWrite(LED,HIGH);
   delay(1000);
   digitalWrite(LED,LOW);
   Serial.print(inches);
   Serial.print("in, ");
   Serial.print(cm);
   Serial.println("cm");
   Serial.println("THE BIN IS FULL! SEND AN E-MAIL USING A 3RD PARTY SOFTWARE LIKE
1SHEELD!!");
  }
  else {
    digitalWrite(LED,LOW);
    Serial.print(inches);
    Serial.print("in, ");
    Serial.print(cm);
    Serial.print("cm");
    Serial.println();
  }
  delay(1000);
}

long microsecondsToInches(long microseconds)
{
  // According to Parallax's datasheet for the PING))), there are
  // 73.746 microseconds per inch (i.e. sound travels at 1130 feet per
  // second).  This gives the distance travelled by the ping, outbound
  // and return, so we divide by 2 to get the distance of the obstacle.
  // See: http://www.parallax.com/dl/docs/prod/acc/28015-PING-v1.3.pdf
  return microseconds / 74 / 2;
}

long microsecondsToCentimeters(long microseconds)
{
  // The speed of sound is 340 m/s or 29 microseconds per centimeter.
  // The ping travels out and back, so to find the distance of the
  // object we take half of the distance travelled.
  return microseconds / 29 / 2;
}
```

Makefile source code:

https://pastebin.com/ZBwvxfUe

```
main compiler
CC := gcc

TARGET1 := ProjectCcode
all: $(TARGET1)

$(TARGET1):
```

```
@echo "Compiling C program"
$(CC) $(CFLAGS) $(TARGET1).c -o $(TARGET1) $(LDFLAGS) -l$(LIB)
```

Text file Source Code- the file we read from and write to (logging implementation) (cfile.txt):

https://pastebin.com/kxqSFmCM

```
1 2 Do you segregate waste?.
3 4 Did you trash non-biodegradable trash today?.
5 6 Did you waste food today?.
1 3 Do you know how many kids die of starvation in Africa?.
1 0 Do you see people throwing food around you everyday?.
1 1 Do you do your bit?.
2 2 Are you happy with your life?.
```

AI Chat Page Source Code (client.html):

https://pastebin.com/QBqVd2Pt

```
<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.
-->
<html>
<head>
    <title>Reduce, Reuse, Recycle- Trash Canada</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
    <script src="nodeServer.js"></script>

    <style type="text/css">
        body { width: 100%; margin: 0 auto; text-align: center; margin-top: 20px;
                background-image:
url("https://orig00.deviantart.net/8538/f/2007/087/4/f/sonny__irobot_by_underexaggerated.jpg");
                background-size : 1000px;
                }
        /*div {  position: absolute; }*/
        input { width: 400px; }
        .button {
                display: inline-block;
                padding: 10px 20px;
                font-size: 15px;
                cursor: pointer;
                text-align: center;
                text-decoration: none;
                outline: none;
                color: #4169e1;
                background-color: white;
                border: none;
                border-radius: 12px;
                box-shadow: 0 4px #999;
                }
                .button:hover {background-color: #8470ff}
                .button:active {
                background-color: #8470ff;
                box-shadow: 0 5px #666;
                transform: translateY(4px);
                }
        textarea { width: 30%; background-color: beige; border-color: burlywood; border-width: 10px}

</style>
</head>
<body>
    <div>
                <img src="https://image.freepik.com/free-icon/waste-can-full-of-trash_318-44388.jpg" alt="trash_talk"
style="width:200px;height:100px" align = "left"><br/>
                <textarea id="input"></textarea> <button id="rec" class="button">Speak</button>
        <br/>

                <h1 style="font-size:300%;text-align:center;font-family:Broadway;color:white">Response</h1>
                <br/> <textarea id="response" cols="40" rows="20"></textarea>
        <button class="button"> I'm done! </button>
    </div>


        <script type="text/javascript">



    var accessToken = "09774994a7a3472fa046ae4724745d32";
        var baseUrl = "https://api.api.ai/v1/";

        <!-- -->
        var transcript = ["This is a transcript"];
        <!-- -->

        $(document).ready(function() {
            $("#input").keypress(function(event) {
                if (event.which == 13) {
                    event.preventDefault();
                    send();
                }
            });
            $("#rec").click(function(event) {
                switchRecognition();
            });
        });
        var recognition;
        function startRecognition() {
            recognition = new webkitSpeechRecognition();
            recognition.onstart = function(event) {
                updateRec();
            };
            recognition.onresult = function(event) {
                var text = "";
                for (var i = event.resultIndex; i < event.results.length; ++i) {
                    text += event.results[i][0].transcript;
                }
                setInput(text);
                stopRecognition();
            };
            recognition.onend = function() {
                stopRecognition();
            };
            recognition.lang = "en-US";
            recognition.start();
        }

        function stopRecognition() {
            if (recognition) {
                recognition.stop();
                recognition = null;
            }
            updateRec();
        }
        function switchRecognition() {
            if (recognition) {
                stopRecognition();
            } else {
                startRecognition();
            }
        }
        function setInput(text) {
            $("#input").val(text);
            send();
        }
        function updateRec() {
            $("#rec").text(recognition ? "Stop" : "Speak");
        }
        function send() {
            var text = $("#input").val();
            setResponse("You : " + text);
            $.ajax({
                type: "POST",
                url: baseUrl + "query?v=20150910",
                contentType: "application/json; charset=utf-8",
                dataType: "json",
                headers: {
                    "Authorization": "Bearer " + accessToken
                },
                data: JSON.stringify({ query: text, lang: "en", sessionId: "somerandomthing" }),
                success: function(data) {
                    setResponse("TrashTalk : " + data.result.fulfillment.speech); // Add
                },
                error: function() {
                    setResponse("Internal Server Error");
                }
            });
            setResponse("Loading...");
        }
        function setResponse(val) {
            $("#response").text(val);

             if(val == "Loading...")
            {}
            else
            transcript.push(val); <!-- -->

            speak(val);




        }

        function speak(string){
var utterance = new SpeechSynthesisUtterance();
utterance.voice = speechSynthesis.getVoices().filter(function(voice){return voice.name == "Junior";})[0];
utterance.text = string;
utterance.lang = "en-US";
utterance.volume = 1; //0-1 interval
utterance.rate = 1;
utterance.pitch = 2; //0-2 interval
speechSynthesis.speak(utterance);
}

    /*   var htmlMeinObj = {

        htmlMeinFunc : function()
        {
            var stringContainingFullTranscript = "Start";

            for(int i = 0; i<2;i++)
            {
```

```
            stringContainingFullTranscript = stringContainingFullTranscript + transcript[i];
            }


            alert("sending");
            obj.passed(stringContainingFullTranscript);
        }
        };

        */




        <!-- -->
        <!-- -->

    </script>
</body>
</html>
```

Peer Contribution & Acknowledgment

We would like to extend our gratitude to David and Matt, our ECE 150 TAs and our Instructor Paul Ward. We would also like to thank our classmates who posted the self-same questions we had on Piazza. We extend our heartfelt gratitude to the Claudette Millar Hall Cleaning Staff for sharing their years of knowledge and experience in safe waste management and providing us with a list of commonly generated waste in the urban university sector.

DON'T FORGET THE 3 R'S- REDUCE, REUSE AND RECYCLE!