Journal 0

Jialin Tso

August 27, 2019

1 Introduction

Blind and physically-impaired people struggle with using physical graphing calculators such as TI-84s because they either have a hard time pressing the buttons or cannot see the graphs. An online voice-controlled graphing calculator would enable these users to interact with the device by allowing them to dictate their commands to the program, listen to the calculator's output, and envision graphs based on sound pitches. These tasks require a background script that checks if someone is talking, a mapping of voice commands to their corresponding calculator buttons, a program confirmation of what the user has said, and an implementation of a screen reader.

2 Obstacles

A main obstacle to this project is to enable the calculator to correctly interpret the user's voice and commands because many speech recognition libraries are not very well crafted. I would also have to address the issue of having the calculator speak in a clear and concise manner as not to annoy the user to the point where a physical calculator is better than a voice-controlled one. Additionally, I need to determine shortcuts or hotkey commands of functions that users would utilize frequently and create a complete list of these commands.

3 First Marker of Success

My first marker of success would be for the program to recognize a user's voice and output the correct commands as requested by the user. For example, if he or she says, "fours times the quantity of six plus five," the calculator would output the expression and 44 on the screen and return in a robotic voice, "the answer to four times the quantity of six plus five is forty-four." Another example would be if the user reads off a long list of numbers and when he or she says, "stat, calc, 1-var stats, L1, enter, enter" there should be a list of these 1-var stats on the screen and the calculator would read them out in order.

4   Materials

Programs that I would need include the Wabbitemu calculator to have a calculator emulator that I can map the voice commands on, the Python speech recognition library, Google Proto Buffer to transmit commands between Python and C++, and vOICe's screen reader for the graph generating component of the project.