Bluetooth Headphone Analysis

[0.5 pts] Group members: Bridget LaBonney

[0.5 pts] Goal of the project: The goal of this project was to analyze the Soundcore Quiet 45 headphones to have a deeper understanding of Bluetooth's innerworkings, and to see if these headphones performed any unique actions or were insecure. The program is now able to do analysis on

[4 pts] Details of Code: Provide details of how to run your code and how to reproduce your work, and any other comments that you have about your code.

- **How to run:** What should I type in the terminal to run your code? (Ex: make, python3 code.py) **[+1]**
  1. **Open analysis.py and look for the filename attribute in main()**
  2. **Change the filename to your filepath for the txt file ("c:/user/files/x.txt…")**
  3. **Type python3 analysis.py in terminal**
- **Environment:** Matplotlib, numpy, & python3. I highly recommend that this code should be run on Spyder, as I built it on that environment. This does work on an Linux environment with the proper modules installed.
- **Input/Output:** The input should be a correctly formatted .txt file and after menu navigation, should output the requested graph.
- **Extra comments: Graphs may look different than pictured.**

[3 pts] Details of data: It should be a pointer to data.

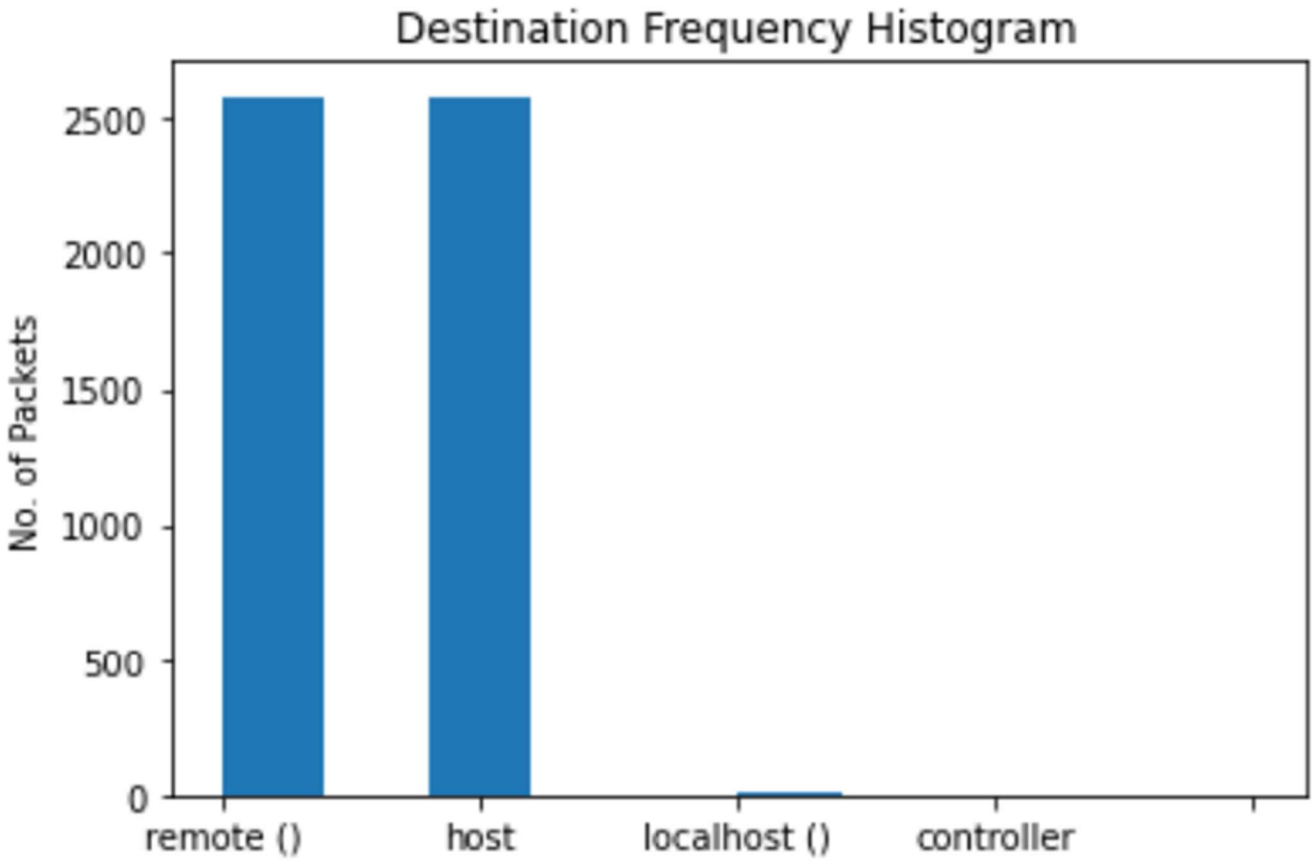
- **Pointer to data:** All packets used are available on this project's github, https://github.com/bridgetlabonney/CPE400HeadphoneProject. I have provided both the raw and the pre-processed packet (x.txt).
- **Cleaning and processing:** After data collection, the packets were then exported to a .csv format and then converted to a tab-delimited.txt file. The data is cleaned further based on user requests in the program.

[2 pts] Methodology: The project was mean to examine how Bluetooth works, so data collection was performed while the headphones were in use (watching a video, etc.) using Wireshark. After filtering to Bluetooth only,

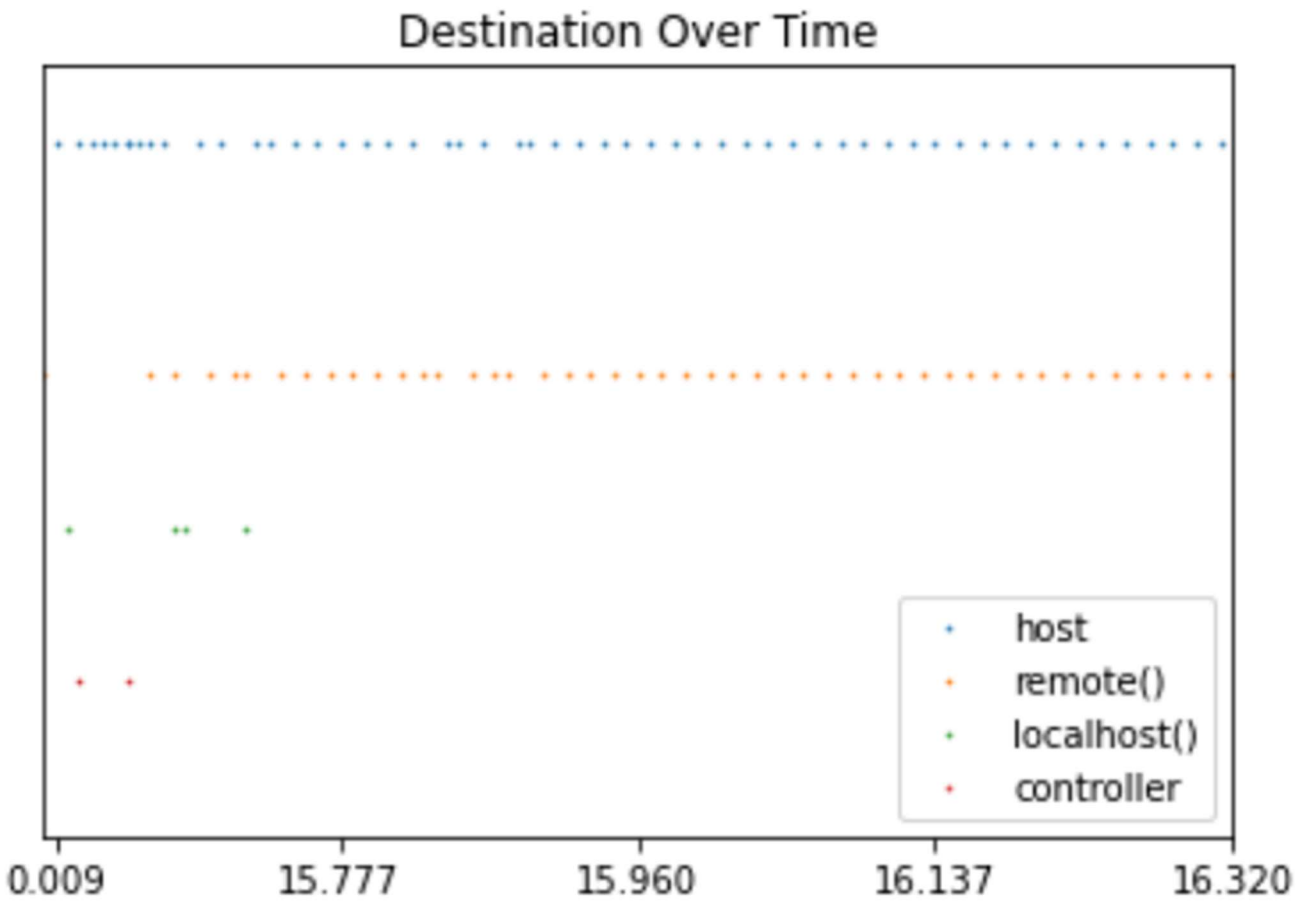
[8 pts] Evaluation results: Please put all evaluation results here.

The program performs 6 different kinds of analysis:

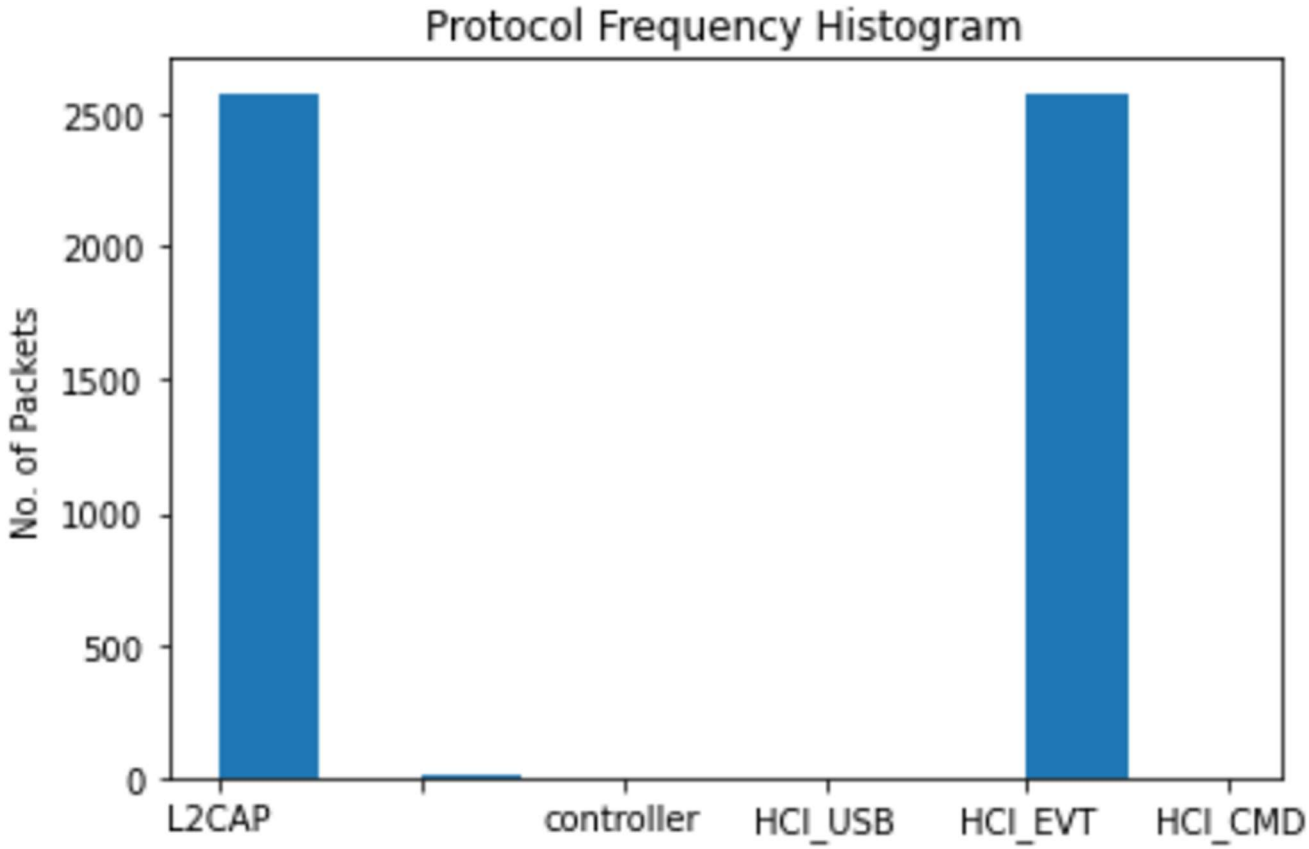
1. Destination Frequency Histogram
   The idea of this analysis was to figure out how often a destination is used, as that destination may see a lot of traffic.



2. Destinations over Time
   Figuring which destinations are used over the course of the use of the headphones. It looks like remote() and host are used for most of the packets after initial setup.

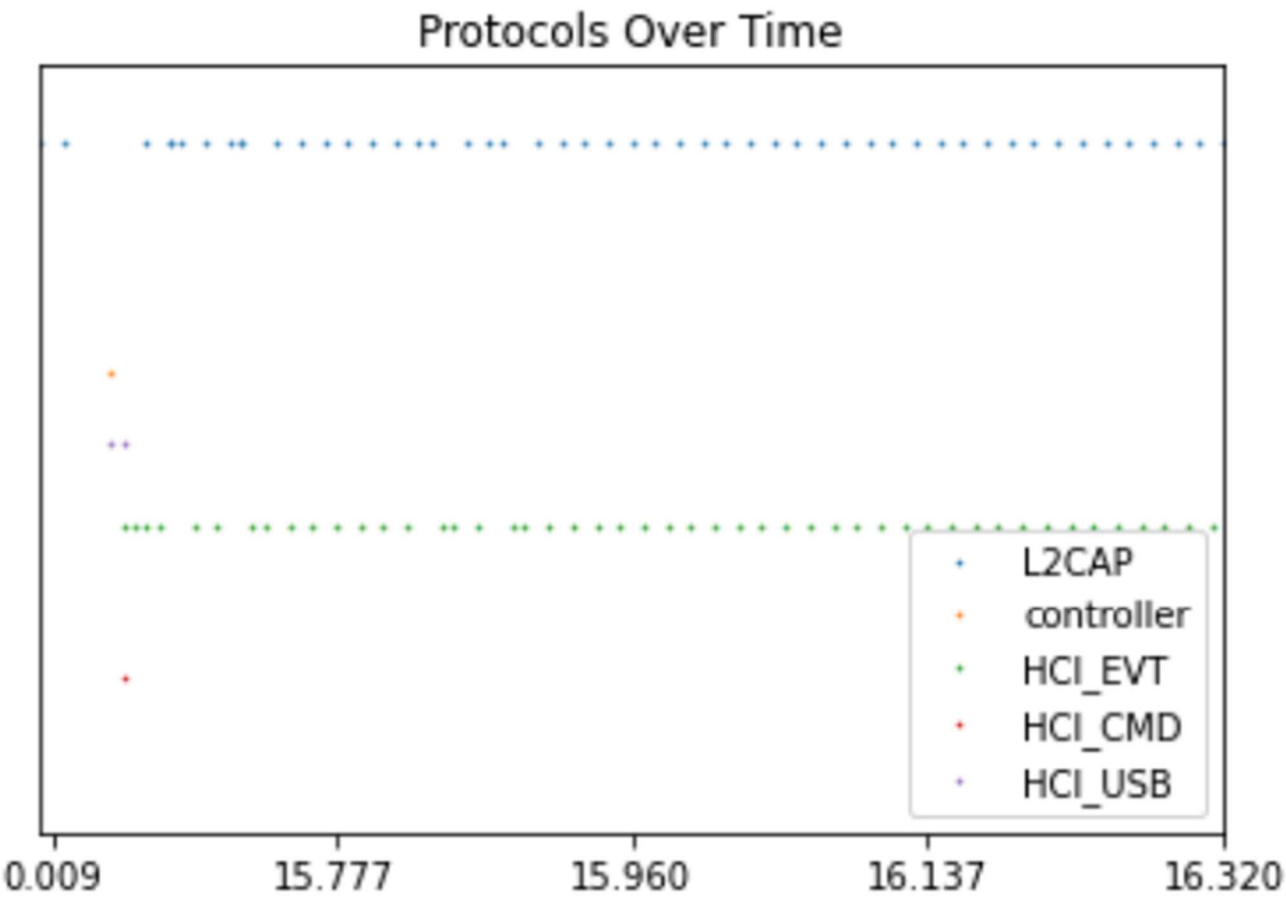


3. Protocol Frequency Histogram
   Similar to analysis 1, protocol frequency is also a part of identifying how these headphones work.



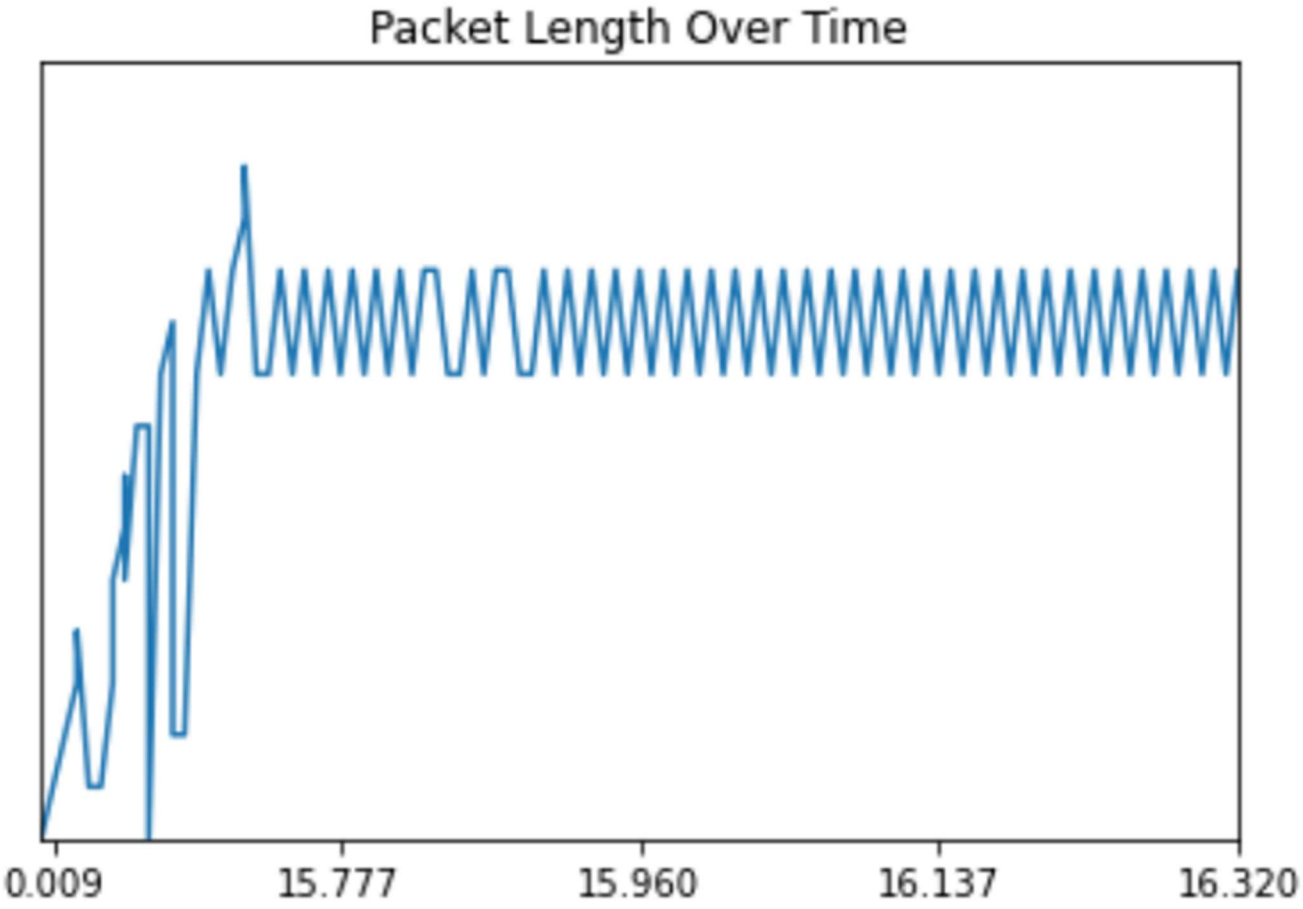
Similar to most of the analysis, it looks like 2 have a higher frequency than most.

4. Protocol over Time

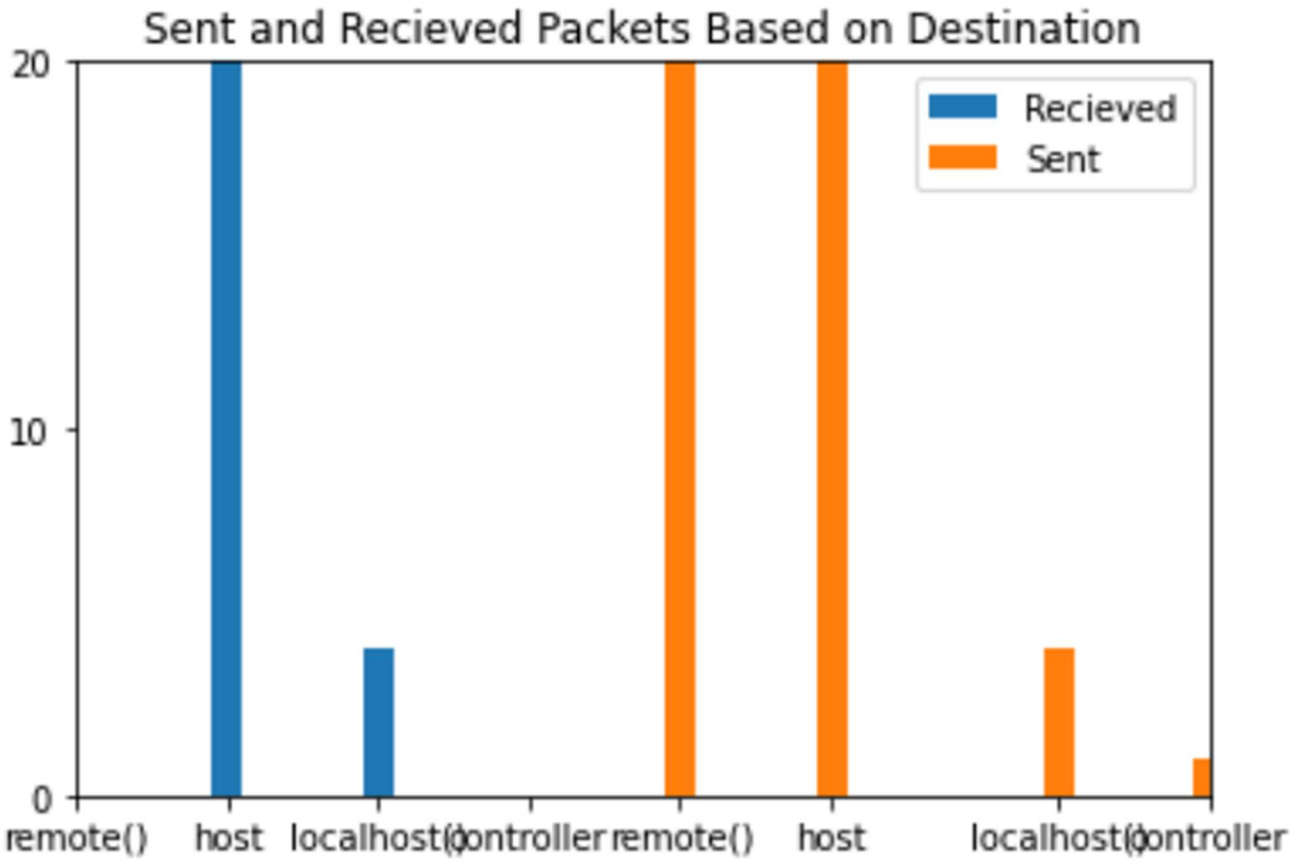


This shows the protocols used, a lot more being utilized during setup and never being seen again.

5. Packet Length over Time
   Packets carry a lot of information. Their size determines how much information is in there, and as we can see on this graph, the length spikes but very quickly evens out over time. This is due to the initial setup procedure as is evidenced by the other graphs.



6. Sent & Received Based on Destination
   As packets are sent back and forth, their destinations are always changing.



Analyzing their destination shows which destinations receive and send the most packets. Remote() never received a packet, showing that that destination is only used for sending out information.

[0.5 pts] Workload: I am the only member in my team.

[0.5 pts] Tools that you used: I used Wireshark to collect the data, Excel to pre-clean, and Spyder to program the analysis program.

[0.5 pts] Challenges: Matplotlib, Python's plotting library, was tough for me to work with, and required a lot of finicking and messing around.

[0.5 pts] Future directions: This project didn't provide me with very many interesting results. It might be wise to consider something that provides a wider range of results, or has a more complicated networking system.