

Zergmap Project Summary

Jonathan Butler

February 10, 2022

1 Project Summary

Per the requirements, we were to develop a program that will work with the Psychic Captures (PCAPs) of network traffic involving the Zerg fell race. When given a PCAP file, Zergmap will determine where the Zerg are located on the battlefield. The program will recommend a list of Zerg to kill in order to form a fully-connected network of the survivors.

2 Challenges

2.1 Using Git as a Group

Prior to this project I have only ever used Git in a solo setting. When using Git during this project, there were a few issues that we experienced along the way (e.g. merge conflicts, merge when we should not have, etc).

2.2 Automated Unit Testing

Being that this is only my second project where I incorporated automated unit testing, I am still not as familiar with this process as I would like to be. During this project I required my teammates, as well as the instructors assistance to successfully put together working unit tests.

2.3 Graphs

This week's instruction was a culmination of the DSA portion of TDQC. Graph's include the use of many of the previously taught data structures. As such, graphs are one of the most complicated and involved data structures that we have used to date. Due to all of the previous data structures still being fairly new to me and then having to incorporate them into the a graph data structure, this proved very difficult. I spent more time on this project trying to understand the basic use / employment of graphs than I did anything else.

3 Successes

3.1 Automated Integration Testing

During this project, old code from decode had to be refactored / optimized. As to mitigate any risks associated with refactoring decode to meet the requirements of Zergmap, I created several bash scripts to automate testing. This was my first time creating bash script, so it could definitely use some refactoring itself. Despite this, the script performed what I needed it to and it helped out immensely during the refactoring process and overall testing of Zergmap.

3.2 Working as Development Team

Prior to working as a group, I was a bit apprehensive. This was due to the fact that I knew using Git with a group could be problematic and personality conflicts could arise. Although there were some growing pains with using Git as a group, it was more beneficial than problematic. Our group also worked very well together, which helped this project go along smoothly.

4 Lessons Learned

4.1 Using Git as a Group

It is clear that using git in a group setting requires clear communication and Git branching strategies. As we continue to work in groups, I will focus on communication and Git branching / tagging strategies with my partners.

4.2 Automated Unit Testing

Based upon my the challenges experienced with automated unit testing, I need to invest time in studying and developing automated unit test suites..