**Luke Hovarter, Nam Le**

**Quad Solver Testing Plan**

Every time a change is made in the program previous tests will be rerun. This regression testing is to make sure that we didn’t add unintentional features while adding intentional features.

Our testing plan is going to use Cunit a port of Junit for C instead of Java for unit testing. This would allow us to check for problems with individual modules. For System testing we plan to run tests using files with expected outputs and inputs and comparing the results.

To test Acceptance, we would create a document that lists all the requirements and review it with the customers. Once we believe the program is finished we will try to use each of the required features, and then have our customer to try each of his requested features to ensure that we understood the requirements as well as included all of them.

To test usability, we will give 4 engineers the completed and compiled program to try to use and see if they understand. 2 of them we will give instruction on how to use it and 2 we will not give any direction to see how they fair with and without training. We would go through our list of use cases created from the requirements for the quad solver from the acceptance testing and see if the users are able to use the program for these cases.

I do not know of any accessibility training I could do for a command line program as it is a rather simple interface that only relies on command line input. The only accessibility I can think of is possibly increasing the font size in the terminal program they are using.

While I do not think that performance testing will be necessary for this type of program, to test performance the program should be tested on the hardware that the engineers plan to use to make sure that is produces answers in a satisfactory amount of time. I cannot imagine it taking a noticeable amount of time.

The testing would be done mostly as black box testing as the program seems simple enough that it does not require the code to be visible to understand what is going on in the program.