QA Assessment

By: Nithushan Selvanesan

Email: nithushan.selvanesan@gmail.com

## Q1. Something is not displaying properly in the browser and you want to learn more about it. How do you go about doing that? (Looking for what tools and specific steps you would take to investigate the error)

## If something is not displaying properly in the browser, the first thing that I would check is the browser console to see if the issue is related to some erroneous JavaScript or CSS/HTML because often times, the issue may be related to either logic or syntax errors. Syntax errors should be fairly easy to fix, however logic errors may not be so intuitive. The next step would be to look at the particular block or <div> that contains the issue and try to pinpoint which HTML or JavaScript file is generating it to see what code is supposed to produce this output. This will give a better understanding on what local and global variables that code depends on to run correctly. Then, I can write unit tests that check those values to make sure that they are what I expect them to be and that they are not undefined or None. I can write unit tests in Python or in JavaScript or using a particular unit testing library. In the worst case, the errors may come from imported libraries which have erroneous code and I may need to update the libraries to the appropriate versions to resolve them.

Q2. Have you ever broken a piece of software? How did you break it? How did you fix it?

I have broken a piece of software before in one of my group projects. I was in charge of working on implementing the Simulated Annealing algorithm for a shortest path delivery application and I ended up breaking the algorithm for the entire project when I accidentally removed a portion of the code that was response for double checking the validity of the new path that was generated. I ended up fixing it by first isolating every major section of the code and then checking the inputs and outputs to ensure that each component was functioning correctly. This helped me determine that the issue with the software was that the validity check at the end of the program was causing the issue.

Q3. Describe your experience with code review.

My experiences with code review often involved me and my partners double checking each other’s work for functionality and clarity. For example, sometimes when we were rushed, we may not write comments or label variables and functions clearly and during these code reviews, we would go through the code and make sure that as we explained our functions, we would also try to fix up any issues with communication. In terms of functionality, there really wasn’t that many issues, but sometimes there may be misunderstandings of how the program should behave during corner cases and it was during these code reviews where we would make sure that all corner cases were handled correctly and without issue. For example, if our code used recursion, then what were the terminating conditions and what were the recursion conditions. Once the functionality and clarity were covered, we would then try to optimize each other’s code. This often involved trying to minimize the number of loops, trying to avoid recursion, using more efficient data structures (i.e hash tables, binary search trees), using dynamic programming, and other techniques to avoid redundant computation.