

# CS31310 Worksheet 2

Daniel Atkinson (daa9)

During the pair programming exercise the main interactions I had with my partner were that of discussion. Discussions involved how to go about tackling the problems outlined in the stories. Once the general idea had been outlined one person sat in the 'drivers' seat and wrote code whilst the other 'navigates'. We did generally both pitch in ideas all the time rather than sticking to strict roles.

This type of interaction seemed to work very well for us as it enabled us to clearly communicate ideas and concerns regarding the assigned task and come to a structured conclusion where we both knew what we had to do to get it finished. This may have felt like it was wasting time at first by having two people working on the same problem and having a large amount of discussion but in actual fact we got the tasks completed and working quite quickly. My assumption here is that we can fill in the blanks for each other instead of trying our own independent implementations and fixing it afterward via a code review and re-writing it.

It is getting a live mini code review all the time.

We did not come across many technical problems, mainly due to my partner being very experienced with Java. We did discuss anything I was slow on or felt needed clarifying such as an appropriate data type for the small currency values involved in the supplied stories.

The practice we followed was for one person to write a test, then the other to implement a solution, run the test and if it passes, great we carry on, if it fails we switch roles and continue in such a manner until the test passes. Then move onto the next task.

I found the constant switching interesting due to the fact that the person typing was not deciding what to do as that was the navigators job.

I believe that me and my partner were an expert-average pairing. The average person being myself, not due to the fact that I do not have programming experience more than my partner was far more experienced with Java, the chosen language for this pair programming session.

This just meant that my partner tutored me in some specifics, which took very little time and increased my knowledge of the language and helped to speed up the later tasks.

Most of the teaching was on usage of the eclipse IDE and using JUnit as I have used these before but several years ago, most of my development is done purely command line these days, it was a nice refresher to GUI development tools.

The live feedback on what I was doing from my more experienced partner was very beneficial and time saving. It increased the quality of my code without having to do any major refactoring as it was being corrected or improved as it was being written.

When writing the tests we wrote them so that they would definitely fail to start with to verify that the unit tests and the testing suite notified us of the failure. After the test failed, we swap roles and write a passing test, again to show that it notifies us of the pass. Finally we implemented the intended solution and ran tests again. This time if it did not pass we would have to switch roles again and discuss why we thought the test did not pass. If this is simple it would get fixed by the current driver and then tests would run again, otherwise the solution would be re-planned and re-implemented with the agreed changes, again running the tests for validation.