COMP2209 – Programming III

Coursework Report

13/01/2022 – Version 1.0

Patrik-Tibor Csanyi – ptc1g20@soton.ac.uk

In this coursework, I completed two out of the 6 challenges: the first and third challenge. The reason why I chose these is simple: the first challenge seemed to be the easiest, and from what I have heard, the second one was incredibly difficult and time consuming. As I got sick during the holidays, and my special consideration application was not approved in time, I decided to attempt the next exercise, which was the third challenge.

For the first challenge my initial approach was to follow the rays and scan the area ahead of each ray. This approach was quite straight forward, however, as I started implementing the idea, I realised that it was significantly more difficult than I initially thought. Then I realised that the location of each atom was available to me, and so it could be significantly easier to keep track of the area of affect of each atom, and check if a ray passes through them. This also allowed me to "label" each of the affected grid points, as a point where the ray is "Absorbed", "Deflected", "Double Deflected", "Reflected" or "Nothing Happens" to it. I tested my code with different board sizes, number of atoms, and atom locations, be that on the edge, next to it, or somewhere deeper inside. These were all the cases I considered, as these all required me to make sure a ray is not affected by an atom outside the board.

For the third challenge my idea was to first convert the expressions into lambda expressions with "printLambda" and then do the Scott encodings. Here, I believe I have to mention a source that I used. During my time trying to figure out how to turn the expressions given into lambda expressions, I was looking for clues and hints online, as I ran out of ideas. Then I stumbled upon this link: https://stackoverflow.com/questions/59338968/printing-lambda-expressions-in-haskell , which had an exact solution to my problem. I was not looking for a solution, just ideas to get me started, however, after I saw this page, I could not think of any other way of solving the problem and ended up implementing it into my own solution. I tested my code by manually calculating different lambda expression and seeing if "prettyPrint" would give me same outcome and if it would transform any suitable lambda expressions to natural numbers