Kenta Hasui

CS331

Assignment 5 Report

For this assignment, we finally got to play around with kroc. Like other programming languages, kroc had many quirks and unique features that I had to familiarize myself with. Working through these issues was a frustrating, but ultimately valuable, learning experience. It was immensely helpful that setting up the environment was so simple - I was able to spend more of my time figuring out the language than running scripts or downloading files.

The first part of the assignment, q1, actually gave me the most trouble. I based the design of processes S0 and S1 on the process numbers, from the occam slides. I simply initialized prefix of S0 to 0 and S1 to 1, and included 2 succ processes within each (to increment the count by 2 each time). I was shocked at how simple building the process became by drawing a diagram. This type of composition is very hard in a language like Java, but was straightforward for kroc. However, I ran into a wall when trying to design process "Alternate." At first, I had two "ID" processes within alternate operating sequentially, reading from each input channel and outputting to the out channel. But when I tested this in the main process (q1), I found that it only output one value for each run through the program. It was not alternating the input at all! After a lot of banging my head against the wall and trying various combinations of processes, I eventually decided to simplify the design. Instead of trying to compose higher-order processes, I composed basic read and write processes. This fixed my problem, and the rest of the q1 assignment was straightforward.

Part 2 of the assignment, q2, gave me a fair share of problems too. I had to realize that the scope of variables only extends to the next statement! The same holds for while loops: if you want to have more than one statement execute inside a while loop, you must enclose it within a SEQ or a PAR. The minus process was just the plus process with subtraction instead of addition. When I designed the differentiate process, however, I chose the wrong combination of individual processes. I initially composed minus, delta, and tail. However, this removed the 0 from the output stream, because the 0 was "absorbed" by the tail process. Thus I realized that I needed to switch the tail to a prefix, and switch the order of subtractions. This worked like a charm, and my process worked as it should!

Overall, this assignment was very fun and fulfilling. Learning a new language is always quite tricky, especially one as unconventional as kroc. But it was rewarding finally get my processes to work. Most of the time, the solution to an issue I was having was to simplify my design! I could see how powerful this type of composition could be if I had more time to get used to the syntax.