

Homework 06

April 10, 2020

Problem 5.22: Let the new rules for the `par` command be defined as such:

$$\begin{array}{c}
 \frac{\langle S'_x, s \rangle \Rightarrow s'}{[PAR1_{SSS}]} \\
 \langle S'_x; \text{skip par skip}, s \rangle \Rightarrow s' \\
 \\
 \frac{\langle S'_y, s \rangle \Rightarrow s'}{[PAR2_{SSS}]} \\
 \langle \text{skip par } S'_y; \text{skip}, s \rangle \Rightarrow s' \\
 \\
 \langle S'_1; S''_1 \text{ par } S'_2; S''_2, s \rangle \Rightarrow \langle S'_1; (S''_1 \text{ par } S'_2; S''_2) \text{ or } S'_2; (S'_1; S''_1 \text{ par } S''_2), s \rangle [PAR3_{SSS}] \\
 \\
 \frac{S_1 \models S'_1; S''_1 \text{ and } S_2 \models S'_2; S''_2}{[PAR4_{SSS}]} \\
 \langle S_1 \text{ par } S_2, s \rangle \Rightarrow \langle S'_1; S''_1 \text{ par } S'_2; S''_2, s \rangle
 \end{array}$$

Where \models is a relationship that splits the elements of a statement into its sequential parts such that S_1 can be represented as S'_1 and S''_1 where S'_1 is the first sequential operation in S_1 and S''_1 is the rest of the statement. If there is only a single operation left in S_1 , S'_1 takes that operation and S''_1 gets the operation `skip`.

This set of rules allows us to interleave all of the possible orders of the `par` command and then non-deterministically choose the path by leveraging the `or` operator. $PAR4_{SSS}$ splits up the statements into their sequential parts using the defined \models operation, $PAR3_{SSS}$ recursively splits up the sequenced commands into their interleaved form, and $PAR1_{SSS}$ and $PAR2_{SSS}$ provide a base case to end the recursive interleaving of $PAR3_{SSS}$.