Brian Rodriguez
U0853503

Quiz 9a

9a:

The assigned reading provides a very interesting way of detecting data-races and presents a method of data-race checking that I have never seen before. I say that I have never seen this sort of method because I have never seen discrete mathematical proofs involved when solving a data-race problem. It makes sense why the author does this sort of thing and I know that discrete mathematics is used in all types of engineering but nevertheless it is an interesting way to present this information.

The author provides two precursor sections that give backbone to presenting the data-race checker. He explains a type of modeling method that uses a DAG data-structure. He presents a fork-join graph that uses closed, nestable fork-join constructs; fork operations are met with their corresponding joins. The author also provides a labeling scheme called Offset-Span labeling. The labeling scheme contains information that identifies its position in its fork-join graph. Comparing the labels, one can deduce their concurrency relationship.

After the precursor sections, the author goes into detecting the data-races with these modeling constructs. Detecting data-races on the fly is quiet interesting, in order for this process to begin a program transformer must keep an access history storage for each shared variable with a reference endpoint of a dependence carried in the parallel construct. The transformer deals with a monitoring protocol that inspects and updates the variable's history respectively. Transformers must also insert statements that let the thread calculate a label that reflects its concurrency relation to other threads. This allows for the monitoring protocol to detect data-races at runtime.