ORACLE GENERATION QUESTIONS & ANSWERS

RON ALFORD

Problem 1. Consider the following temporal logic statements.

(1) $\Diamond A$
(2) $\Box(\Diamond A)$

where $\Diamond$ means 'eventually', and $\Box$ means 'always.' In class, we discussed how these two have different meanings. Explain why.

Consider two state sequences:

(1) $BBBABBBB$
(2) $BBBABBBA$

The first statement ($\Diamond A$) states that 'A' must eventually hold. This is the case in both sequences, since 'A' is true in the fourth state.

In plain english, the second statement ($\Box(\Diamond A)$) states that at any given point, 'A' will eventually hold true. By the fifth state of the first sequence, there are no more 'A', and so the second statement does not hold for that sequence. In contrast, the second sequence ends with 'A'

Problem 2. Why would you want to examine a program's execution trace instead of simply verifying the output (which may be easier?)

Trace validation can be used to enforce behavior of programs that might not have an immediate impact on the output. For example, one execution trace constraint might be that every file opened is also closed. Other examples include checking for matched semaphore locks/unlocks and insuring all database transactions are committed or rolled back.

Problem 3. How can pre- and postconditions be used to simplify the oracle generation process? What are the risks?

Pre- and postconditions decouple checking the inputs and outputs to each function from the rest of the test case oracle.

Since postcondition checks may need to compare the state of the program before and after execution of the function, there is memory overhead to using them. This may cause an unacceptable performance hit in some cases (especially recursive functions).

Date: May 7, 2008.