CS266 - Formal Specification and Verification

HW #6 -- Winter 2009

More ASLAN Specifications and Proof Obligations

Due: Thursday 19 FEB 09

Part I - Exercise to Complete Existing Specification

Consider the following problem:

A farmer has to carry a goat, a cabbage, and a wolf from one side of the river to the other. He has a boat to do so, but the boat cannot carry more than three items, whether man, goat, cabbage, or wolf. If he leaves the goat and the cabbage on the same side of the river or in the boat without being present, then the goat will eat the cabbage. Similarly, the wolf will eat the goat if the farmer is not present.

An incomplete ASLAN specification for the problem is presented below. You are to fill in the invariant and the necessary pre and post conditions for the single transition Transport. Your invariant should express the critical constraints of the problem. You are not to solve the problem. Therefore, your entry assertion only need express the obvious requirements, such as boat capacity.

What you are to turn in to me for this question is the listings of the .out file corresponding to the question.

```
SPECIFICATION Boat
LEVEL Top_Level

TYPE
    River_Bank IS (East, West),
    Occupant_Type IS (Farmer, Goat, Cabbage, Wolf),
    Occupant_Set IS SET OF Occupant_Type

VARIABLE
    Location(Occupant_Type): River_Bank,
    Boat_Location: River_Bank

CONSTANT
    Cardinality(Occupant_Set): Integer
        /* Indicates the number of elements in the set */

INITIAL
     FORALL o:Occupant_Type (Location(o) = East)
    & Boat_Location = East

INVARIANT
     *** FILL THIS IN ***

TRANSITION Transport(Passengers:Occupant_Set)
     /* This transition moves passengers from one side of the river to the other in the boat  */
 ENTRY
     *** FILL THIS IN ***
 EXIT
     *** FILL THIS IN ***

END Top_Level
END Boat
```

Part II - Checking the Correctness of Aslan Theorem Generation

1) You are to use the Aslan processor to generate the conjectures for the top-level specification of the secure terminal example (using file secTLS) and to generate the conjectures for the second-level specification of the secure terminal example (using file sec2LS).

2) For the Initial Conditions, Connect_To, and Send_Data conjectures for the top level specification you are to mark the constituent parts using the notation presented on page 24 of the Aslan User's Manual.
Also, indicate if any of the conjectures generated are *incorrect*.

3) For the Initial Conditions, Connect_To (for both mapping Connect_To_High and mapping Connect_To_Low), and Review_Data (for the mapping to Accept *ONLY*) conjectures for the second level specification you are to mark the constituent parts using the notation presented on pages 25-27 of the Aslan User's Manual.
Again, indicate if any of the conjectures generated are *incorrect*.

DO NOT print out all of the conjectures only those parts that you are going to mark.
The Aslan processor output for the secTLS specification is 23 pages and for the sec2LS specification is 65 pages.