Solution proposal to selected exercises from Lecture 5

Jens Chr. Godskesen
IT University of Copenhagen
Exercise 3, (Semaphore and Dining Philosophers)

agent Semaphore0(u,d) = ...

agent Forks(g1,g2,g3,g4,g5,p1,p2,p3,p4,p5) = ...

agent Phil(getL,putL,getR,putR,eat,u,d) =
  'd.'getL.'getR.eat.'putL.'putR.'u.Phil<getL,putL,getR,putR,eat,u,d>
agent Phils(g1,g2,g3,g4,g5,p1,p2,p3,p4,p5,e1,e2,e3,e4,e5,u,d) =
  Phil(g1,p1,g5,p5,e1,u,d) | Phil(g2,p2,g1,p1,e2,u,d)
  | Phil(g3,p3,g2,p2,e3,u,d) | Phil(g4,p4,g3,p3,e4,u,d)
  | Phil(g5,p5,g4,p4,e5,u,d)
agent Table(e1,e2,e3,e4,e5) =
  (~g1,g2,g3,g4,g5,p1,p2,p3,p4,p5)
  ( Forks(g1,g2,g3,g4,g5,p1,p2,p3,p4,p5)
    | (~u,d)( Semaphore4(u,d) |
      Phils(g1,g2,g3,g4,g5,p1,p2,p3,p4,p5,e1,e2,e3,e4,e5,u,d) ) )
Exercise 4, (Safety Property: Dining Philosophers)

State that two neighbouring philosophers, say 1 and 2, cannot eat at the same time. MWB confirms this as demonstrated by

MWB>prove Table<e1,e2,e3,e4,e5>
  nu X.(([e1]FF | [e2]FF)
Model Prover says: YES!

The more general variant stating that no two neighbouring philosophers eat concurrently is specified (and proved) by

MWB>prove Table<e1,e2,e3,e4,e5>
Model Prover says: YES!
Exercise 5, (Weak Safety Property)

\[ \nu X. (\phi \land ( \bigwedge_{a \in A} [a] \phi \lor \bigvee_{a \in A} (a) X)) \]

means that \( \phi \) holds in the current state and

- either there are no transitions leaving the current state (the path ends),

- or there is (at least one) transition leaving the current state where the formula holds recursively