```
- MODULE DieHarder
EXTENDS Integers, TLC, GCD
Min(m, n) \stackrel{\triangle}{=} \text{IF } m < n \text{ THEN } m \text{ ELSE } n
CONSTANTS Goal, Jugs, Capacity
Assume \land Goal \in Nat
           \land Capacity \in [Jugs \rightarrow Nat \setminus \{0\}]
 ************************
--algorithm DieHarder{
  variables injug = [j \in Jugs \mapsto 0];
  { while ( TRUE )
      { either with (j \in Jugs) fill jug j
                    \{ injug[j] := Capacity[j] \}
        \mathbf{or}
                 with (j \in Jugs) empty jug j
                    \{ injug[j] := 0 \}
                 with (j \in Jugs, k \in Jugs \setminus \{j\}) pour from jug j to jug k
        \mathbf{or}
                    { with ( poured =
                                  Min(injug[j] + injug[k], Capacity[k]) - injug[k])
                         \{ injug[j] := injug[j] - poured \|
                           injug[k] := injug[k] + poured;
                     }
       }
   }
 BEGIN TRANSLATION
VARIABLE injug
vars \stackrel{\triangle}{=} \langle injug \rangle
Init \triangleq
          Global variables
           \land injug = [j \in Jugs \mapsto 0]
Next \triangleq \lor \land \exists j \in Jugs :
                   injug' = [injug \ EXCEPT \ ![j] = Capacity[j]]
           \lor \land \exists j \in Jugs :
                   injug' = [injug \text{ EXCEPT } ![j] = 0]
           \lor \land \exists j \in Jugs :
                   \exists k \in Jugs \setminus \{j\}:
                     LET poured \stackrel{\triangle}{=} Min(injug[j] + injug[k], Capacity[k]) - injug[k]IN
                       injug' = [injug \ EXCEPT \ ![j] = injug[j] - poured,
                                                    ![k] = injug[k] + poured]
```

 $Spec \stackrel{\triangle}{=} Init \wedge \Box [Next]_{vars}$

END TRANSLATION

- $\backslash * \ {\it Modification History}$
- * Last modified Wed Jun 04 16:38:40 CST 2014 by yaojingguo * Created Wed Jun 04 14:07:37 CST 2014 by yaojingguo