

Algorithm *DieHarder*

MODULE *DieHarder*

EXTENDS *Integers*

$\text{Min}(m, n) \triangleq \text{IF } m < n \text{ THEN } m \text{ ELSE } n$

CONSTANTS *Goal, Jugs, Capacity*

ASSUME $\wedge \text{Goal} \in \text{Nat}$
 $\wedge \text{Capacity} \in [\text{Jugs} \rightarrow \text{Nat} \setminus \{0\}]$

(*****)

```
--algorithm DieHarder {  
  variable injug = [  $j \in \text{Jugs} \mapsto 0$  ] ;  
  { while ( TRUE )  
    { either with (  $j \in \text{Jugs}$  )    fill jug j  
      {  $\text{injug}[j] := \text{Capacity}[j]$  }  
    or    with (  $j \in \text{Jugs}$  )    empty jug j  
      {  $\text{injug}[j] := 0$  }  
    or    with (  $j \in \text{Jugs}, k \in \text{Jugs} \setminus \{j\}$  )    pour from jug j to jug k  
      { with ( poured =  
           $\text{Min}(\text{injug}[j] + \text{injug}[k], \text{Capacity}[k]) - \text{injug}[k]$  )  
        {  $\text{injug}[j] := \text{injug}[j] - \text{poured}$  ||  
           $\text{injug}[k] := \text{injug}[k] + \text{poured}$   
        }  
      }  
    }  
  }  
}
```

(*****)