―――――――――――――― MODULE *peggame* ――――――――――――――

EXTENDS *TLC*, *Integers*, *FiniteSets*

VARIABLES *state*, *xx*, *yy*, *dd*

Spots
   The set of $\langle x,\, y \rangle \in \langle 1\,..\,5,\, 1\,..\,5 \rangle$ where $(x + y \leq 6)$
$Spots \;\triangleq\; \{\langle x,\, y \rangle \in \{1,\, 2,\, 3,\, 4,\, 5\} \times \{1,\, 2,\, 3,\, 4,\, 5\} : x + y \leq 6\}$

Init
   state = The set of *Spots* except (4,1)
$Init \;\triangleq\; \land\, state = Spots \setminus \{\langle 4,\, 1 \rangle\} \land xx = -1 \land yy = -1 \land dd = ``"$

$CanJumpUpRight(x,\, y)$
      A function $f(x,\, y)$ returning true if:
        - The tuple $\langle x,\, y \rangle$ is in *Spots*, is in the current state
        - The tuple $\langle x + 1,\, y \rangle$ is in *Spots*, is in the current state
        - The tuple $\langle x + 2,\, y \rangle$ is in *Spots*, is not in the current state
$CanJumpUpRight(x,\, y) \;\triangleq\; \land\, \langle x,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x + 1,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x + 2,\, y \rangle \in (Spots \setminus state)$

$CanJumpDownLeft(x,\, y) \;\triangleq\; \land\, \langle x,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x - 1,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x - 2,\, y \rangle \in (Spots \setminus state)$

$CanJumpUpLeft(x,\, y) \;\triangleq\; \land\, \langle x,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x + 1,\, y - 1 \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x + 2,\, y - 2 \rangle \in (Spots \setminus state)$

$CanJumpDownRight(x,\, y) \;\triangleq\; \land\, \langle x,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x - 1,\, y + 1 \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x - 2,\, y + 2 \rangle \in (Spots \setminus state)$

$CanJumpRight(x,\, y) \;\triangleq\; \land\, \langle x,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x,\, y + 1 \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x,\, y + 2 \rangle \in (Spots \setminus state)$

$CanJumpLeft(x,\, y) \;\triangleq\; \land\, \langle x,\, y \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x,\, y - 1 \rangle \in state$
$\qquad\qquad\qquad\qquad\qquad \land\, \langle x,\, y - 2 \rangle \in (Spots \setminus state)$

$JumpUpRight(x,\, y)$
   The current state, except . . .
      - Minus $\langle x,\, y \rangle$
      - Minus $\langle x + 1,\, y \rangle$
      - Plus $\langle x + 2,\, y \rangle$

$JumpUpRight(x,\ y)\ \ \ \triangleq\ (((state \setminus \{\langle x,\ y\rangle\}) \setminus \{\langle x+1,\ y\rangle\}) \cup \{\langle x+2,\ y\rangle\})$
$JumpDownLeft(x,\ y)\ \triangleq\ (((state \setminus \{\langle x,\ y\rangle\}) \setminus \{\langle x-1,\ y\rangle\}) \cup \{\langle x-2,\ y\rangle\})$

$JumpUpLeft(x,\ y)\ \ \ \ \ \triangleq\ (((state \setminus \{\langle x,\ y\rangle\}) \setminus \{\langle x+1,\ y-1\rangle\}) \cup \{\langle x+2,\ y-2\rangle\})$
$JumpDownRight(x,\ y)\ \triangleq\ (((state \setminus \{\langle x,\ y\rangle\}) \setminus \{\langle x-1,\ y+1\rangle\}) \cup \{\langle x-2,\ y+2\rangle\})$

$JumpRight(x,\ y)\ \triangleq\ (((state \setminus \{\langle x,\ y\rangle\}) \setminus \{\langle x,\ y+1\rangle\}) \cup \{\langle x,\ y+2\rangle\})$
$JumpLeft(x,\ y)\ \ \ \triangleq\ (((state \setminus \{\langle x,\ y\rangle\}) \setminus \{\langle x,\ y-1\rangle\}) \cup \{\langle x,\ y-2\rangle\})$

Win returns TRUE if there is one peg left
$Win\ \triangleq\ Cardinality(state) = 1$

Next

The next $state(s)$ are those which are the $JumpUp(\ .\ .\ )$ of the current state
$Next\ \triangleq\ \exists\,\langle x,\ y\rangle \in Spots :$
$\qquad\qquad \wedge\ xx' = x \wedge yy' = y$
$\qquad\qquad \wedge\ \ \vee\ CanJumpUpRight(x,\ y)\qquad \wedge\ state' = JumpUpRight(x,\ y)\qquad \wedge\ dd' = \text{``UpRight''}$
$\qquad\qquad\qquad \vee\ CanJumpDownLeft(x,\ y)\quad \wedge\ state' = JumpDownLeft(x,\ y)\ \ \wedge\ dd' = \text{``DownLeft''}$
$\qquad\qquad\qquad \vee\ CanJumpUpLeft(x,\ y)\qquad \wedge\ state' = JumpUpLeft(x,\ y)\qquad \wedge\ dd' = \text{``UpLeft''}$
$\qquad\qquad\qquad \vee\ CanJumpDownRight(x,\ y) \wedge\ state' = JumpDownRight(x,\ y) \wedge\ dd' = \text{``DownRight''}$
$\qquad\qquad\qquad \vee\ CanJumpRight(x,\ y)\qquad \wedge\ state' = JumpRight(x,\ y)\qquad \wedge\ dd' = \text{``Right''}$
$\qquad\qquad\qquad \vee\ CanJumpLeft(x,\ y)\qquad\ \ \wedge\ state' = JumpLeft(x,\ y)\qquad\ \ \wedge\ dd' = \text{``Left''}$

\ * Modification History
\ * Last modified *Tue Mar* 12 10:23:54 *EDT* 2019 by jay
\ * Created Sun *Mar* 10 00:12:41 *EST* 2019 by jay