;; GDL code

;; Don't forget to upload description of the game as:

;; EsCat.txt and/or

;; EsCat.pdf and/or

;; EsCat.html

;;

;; Paste your code here and click on "Update" to make the game globally visible

;;

;; Good luck!

;players p1, p2

(role p1)

(role p2)

(init (roleOfPlayer p1 +)) ;(+) is the catcher

(init (roleOfPlayer p2 \*)) ;(\*) is the escaper

;;;;;;;;;;;;;;;;;

;;;;;cells

;;;;;;;;;;;;;;;;;

(init (cell 1 1 +))

(init (cell 1 2 b))

(init (cell 1 3 b))

(init (cell 1 4 b))

(init (cell 1 5 b))

(init (cell 1 6 b))

(init (cell 1 7 +))

(init (cell 2 1 b))

(init (cell 2 2 b))

(init (cell 2 3 b))

(init (cell 2 4 b))

(init (cell 2 5 b))

(init (cell 2 6 b))

(init (cell 2 7 b))

(init (cell 3 1 b))

(init (cell 3 2 b))

(init (cell 3 3 b))

(init (cell 3 4 b))

(init (cell 3 5 b))

(init (cell 3 6 b))

(init (cell 3 7 b))

(init (cell 4 1 b))

(init (cell 4 2 b))

(init (cell 4 3 b))

(init (cell 4 4 \*))

(init (cell 4 5 b))

(init (cell 4 6 b))

(init (cell 4 7 b))

(init (cell 5 1 b))

(init (cell 5 2 b))

(init (cell 5 3 b))

(init (cell 5 4 b))

(init (cell 5 5 b))

(init (cell 5 6 b))

(init (cell 5 7 b))

(init (cell 6 1 b))

(init (cell 6 2 b))

(init (cell 6 3 b))

(init (cell 6 4 b))

(init (cell 6 5 b))

(init (cell 6 6 b))

(init (cell 6 7 b))

(init (cell 7 1 +))

(init (cell 7 2 b))

(init (cell 7 3 b))

(init (cell 7 4 b))

(init (cell 7 5 b))

(init (cell 7 6 b))

(init (cell 7 7 +))

(init (count 1))

(init (step 1))

(init (round 1))

(init (catchedThrough 0))

(init (control p2))

;;;;;;;;;;;;;;;;;;;

;;;updates of cells

;;;;;;;;;;;;;;;;;;;

(<= (next (cell ?x ?y b))

(does ?player (move ?x ?y ?m ?n)))

(<= (next (cell ?x ?y ?state))

(true (roleOfPlayer ?player ?state))

(does ?player (move ?m ?n ?x ?y)))

(<= (next (cell ?x3 ?y3 ?state))

(true (cell ?x3 ?y3 ?state))

(does ?player (move ?x1 ?y1 ?x2 ?y2))

(distinctCell ?x1 ?y1 ?x3 ?y3)

(distinctCell ?x2 ?y2 ?x3 ?y3))

(<= (next (cell ?x ?y ?state))

catched

(init (cell ?x ?y ?state)))

(<= (next (cell ?x ?y ?state))

startNewRound

(init (cell ?x ?y ?state)))

(<= (next (count ?n)) ;in every round counts 1 -> 25

(not catched)

(true (count ?m))

(pls ?m ?n)

(distinct ?m 25))

(<= (next (count 0)) ;zero to begin other new one

catched)

(<= (next (count 0)) ;zero to begin other new one

(true (count 25)))

(<= (next (round 2))

startNewRound)

(<= (next (round ?x)) ;states the current round (1st or 2nd)

(true (round ?x))

(not startNewRound))

;(step) counts the moves played till the current state

(<= (next (step ?n))

(true (step ?m))

(pls ?m ?n))

;;;;;;;;control;;;;;;;;;

(<= (next (control p1))

(true (control p2)))

(<= (next (control p2))

(true (control p1)))

;switching roles of players at step 25

(<= (next (roleOfPlayer ?player ?role))

(not startNewRound)

(true (roleOfPlayer ?player ?role)))

(<= (next (roleOfPlayer p1 \*))

startNewRound)

(<= (next (roleOfPlayer p2 +))

startNewRound)

(<= (next (catchedThrough 1)) ;1 when getting catched in the 1st round

catched

(true (round 1))

(true (catchedThrough 0)))

(<= (next (catchedThrough ?x)) ;Otherwise stays the same

(true (catchedThrough ?x)))

;;;;;legal moves ;;;;;;

(<= (legal ?player (move ?x ?y ?newX ?newY)) ;diagonal for both

(not startNewRound)

(true (control ?player))

(true (roleOfPlayer ?player ?state))

(true (cell ?x ?y ?state))

(true (cell ?newX ?newY b))

(adjacentD ?x ?y ?newX ?newY))

(<= (legal ?player (move ?x ?y ?newX ?newY)) ;one straight move for +

(not startNewRound)

(true (cell ?x ?y +))

(true (cell ?newX ?newY b))

(adjacent ?x ?y ?newX ?newY)

(true (roleOfPlayer ?player +))

(true (control ?player)))

(<= (legal ?player (move ?x ?y ?newX ?y)) ;two straight to the right for +

(not startNewRound)

(true (cell ?x ?y +))

(true (cell ?newX ?y b))

(pls ?x ?m)

(pls ?m ?newX)

(true (roleOfPlayer ?player +))

(true (control ?player)))

(<= (legal ?player (move ?x ?y ?newX ?y)) ;two straight to the left for +

(not startNewRound)

(true (roleOfPlayer ?player +))

(true (cell ?x ?y +))

(true (cell ?newX ?y b))

(sb ?x ?m)

(sb ?m ?newX)

(true (control ?player)))

(<= (legal ?player (move ?x ?y ?x ?newY)) ;two straight up for +

(not startNewRound)

(true (cell ?x ?y +))

(true (roleOfPlayer ?player +))

(true (cell ?x ?newY b))

(pls ?y ?m)

(pls ?m ?newY)

(true (control ?player)))

(<= (legal ?player (move ?x ?y ?x ?newY)) ;two straight down for +

(not startNewRound)

(true (cell ?x ?y +))

(true (cell ?x ?newY b))

(sb ?y ?m)

(sb ?m ?newY)

(true (roleOfPlayer ?player +))

(true (control ?player)))

(<= (legal p1 noop)

(not startNewRound)

(true (control p2)))

(<= (legal p2 noop)

(not startNewRound)

(true (control p1)))

;;both playing no move when turning to new round

(<= (legal p2 noop)

startNewRound)

(<= (legal p1 noop)

startNewRound)

(<= (legal p2 noop)

catched)

;;;;;;;;goals;;;;;;;

(<= (goal p1 100)

(true (catchedThrough 1))

(not catched))

(<= (goal p2 0)

(true (catchedThrough 1 ))

(not catched))

(<= (goal p2 100)

(true (catchedThrough 0))

catched)

(<= (goal p1 0)

(true (catchedThrough 0))

catched)

;;tie getting 50

(<= (goal p2 50)

(true (catchedThrough 0))

(not catched))

(<= (goal p2 50)

(true (catchedThrough 1))

catched)

(<= (goal p1 50)

(true (catchedThrough 0))

(not catched))

(<= (goal p1 50)

(true (catchedThrough 1))

catched)

;;;;;;;;terminal;;;;;;;

(<= terminal

(true (round 2))

(true (step 52)))

(<= terminal

(true (round 2))

catched)

;;;;;;;;;helper predicated;;;;;;

(<= catched

(true (cell 1 1 \*))

(true (cell 2 2 +)))

(<= catched

(true (cell 1 7 \*))

(true (cell 2 6 +)))

(<= catched

(true (cell 7 1 \*))

(true (cell 6 2 +)))

(<= catched

(true (cell 7 7 \*))

(true (cell 6 6 +)))

(<= catched

(true (cell ?x 7 \*))

(not\_border ?x)

(diagonalDownLeft ?x 7 ?m3 ?n3)

(diagonalDownRight ?x 7 ?m4 ?n4)

(true (cell ?m3 ?n3 +))

(true (cell ?m4 ?n4 +)))

(<= catched

(true (cell ?x 1 \*))

(not\_border ?x)

(diagonalUpLeft ?x 1 ?m3 ?n3)

(diagonalUpRight ?x 1 ?m4 ?n4)

(true (cell ?m3 ?n3 +))

(true (cell ?m4 ?n4 +)))

(<= catched

(true (cell 1 ?y \*))

(not\_border ?y)

(diagonalUpRight 1 ?y ?m3 ?n3)

(diagonalDownRight 1 ?y ?m4 ?n4)

(true (cell ?m3 ?n3 +))

(true (cell ?m4 ?n4 +)))

(<= catched

(true (cell 7 ?y \*))

(not\_border ?y)

(diagonalUpLeft 7 ?y ?m3 ?n3)

(diagonalDownLeft 7 ?y ?m4 ?n4)

(true (cell ?m3 ?n3 +))

(true (cell ?m4 ?n4 +)))

(<= catched

(true (cell ?x ?y \*))

(diagonalUpLeft ?x ?y ?m1 ?n1)

(diagonalUpRight ?x ?y ?m2 ?n2)

(diagonalDownLeft ?x ?y ?m3 ?n3)

(diagonalDownRight ?x ?y ?m4 ?n4)

(true (cell ?m1 ?n1 +))

(true (cell ?m2 ?n2 +))

(true (cell ?m3 ?n3 +))

(true (cell ?m4 ?n4 +)))

(<= (distinctCell ?x ?y ?m ?n)

(distinct ?x ?m))

(<= (distinctCell ?x ?y ?m ?n)

(distinct ?y ?n))

(<= (diagonalUpLeft ?x1 ?y1 ?x2 ?y2)

(pls ?y1 ?y2)

(sb ?x1 ?x2))

(<= (diagonalUpRight ?x1 ?y1 ?x2 ?y2)

(pls ?y1 ?y2)

(pls ?x1 ?x2))

(<= (diagonalDownLeft ?x1 ?y1 ?x2 ?y2)

(sb ?y1 ?y2)

(sb ?x1 ?x2))

(<= (diagonalDownRight ?x1 ?y1 ?x2 ?y2)

(sb ?y1 ?y2)

(pls ?x1 ?x2))

(<= startNewRound

(true (count 0)))

(<= (not\_border ?value)

(distinct ?value 1)

(distinct ?value 7))

(<= (adjacent ?x1 ?y ?x2 ?y)

(adjacent ?x1 ?x2)) ; horizontal

(<= (adjacent ?x ?y1 ?x ?y2)

(adjacent ?y1 ?y2)) ; vertical

(<= (adjacentD ?x1 ?y1 ?x2 ?y2)

(adjacent ?x1 ?x2) ; diagonal

(adjacent ?y1 ?y2))

(<= (adjacent ?a ?b)

(pls ?a ?b))

(<= (adjacent ?a ?b)

(pls ?b ?a))

;;;;;;;;--;;;;;;;

(<= (sb ?m ?n)

(pls ?n ?m))

;;;;;;;++;;;;;;;

(pls 0 1)

(pls 1 2)

(pls 2 3)

(pls 3 4)

(pls 4 5)

(pls 5 6)

(pls 6 7)

(pls 7 8)

(pls 8 9)

(pls 9 10)

(pls 10 11)

(pls 11 12)

(pls 12 13)

(pls 13 14)

(pls 14 15)

(pls 15 16)

(pls 16 17)

(pls 17 18)

(pls 18 19)

(pls 19 20)

(pls 20 21)

(pls 21 22)

(pls 22 23)

(pls 23 24)

(pls 24 25)

(pls 25 26)

(pls 26 27)

(pls 27 28)

(pls 28 29)

(pls 29 30)

(pls 30 31)

(pls 31 32)

(pls 32 33)

(pls 33 34)

(pls 34 35)

(pls 35 36)

(pls 36 37)

(pls 37 38)

(pls 38 39)

(pls 39 40)

(pls 40 41)

(pls 41 42)

(pls 42 43)

(pls 43 44)

(pls 44 45)

(pls 45 46)

(pls 46 47)

(pls 47 48)

(pls 48 49)

(pls 49 50)

(pls 50 51)

(pls 51 52)

;;;;;;colors of players' cells

(show \* (show blue \*))

(show + (show red +))