ADT Tetris

Service : Tetris
Type : int, bool
Require : Board, Block

Observators :
getScore : [Tetris] -> int
isFinished : [Tetris] -> bool
isRunning : [Tetris] -> bool
needNext : [Tetris] -> bool
précondition : needText(T) require isRunning(T)
getBoard : [Tetris] -> Board

Constructor :
init : -> [Tetris]

Opérations :
goLeft : [Tetris] -> [Tetris]
précondition : *goLeft*(T) require *isRunning*(T)

goRight : [Tetris] -> [Tetris]
précondition : *goRight*(T) require *isRunning*(T)

goDown : [Tetris] -> [Tetris]
précondition : *goDown*(T) require *isRunning*(T)

rotateLeft : [Tetris] -> [Tetris]
précondition : *rotateLeft*(T) require *isRunning*(T)

rotateRight : [Tetris] -> [Tetris]
précondition : *rotateRight*(T) require *isRunning*(T)

step : [Tetris] -> [Tetris]
précondition : *step*(T) require *isRunning*(T)

next : [Tetris] -> [Tetris]
précondition : *next*(T) require *needNext*(T)

Observations :

- invariants
  getScore(T) >= 0
  isFinished(T) = ¬*isRunning*(T)

- init
  getScore(_init_()) = 0
  getBoard(_init_()) = **Board :**_init_(10,22)
  isRunning(_init_()) = true
  needNext(_init_()) = true

- goLeft
  getScore(_goLeft_(T)) = _getScore_(T)
  needNext(_goLeft_(T)) = false
  isRunning(_goLeft_(T)) = true
  getBoard(_goLeft_(T)) = **Board :**_doLeft_(_getBoard_(T))

- goRight
  getScore(_goRight_(T)) = _getScore_(T)
  needNext(_goRight_(T)) = false
  isRunning(_goRight_(T)) = true
  getBoard(_goRight_(T)) = **Board :**_doRight_(_getBoard_(T))

- goDown
  Board :_isBottom_(_getBoard_(T)) => _getScore_(_goDown_(T)) = _getScore_(T)
  ¬**Board :**_isBottom_(_getBoard_(T)) => _getScore_(_goDown_(T)) = _getScore_(T) + 20 + **Board :**_getNbLastCleaned_(_getBoard_(_goDown_(T)))*50
  ¬**Board :**_isBottom_(_getBoard_(T)) => _needNext_(_goDown_(T)) = true
  isRunning(_goDown_(T)) = true
  getBoard(_goDown_(T)) = **Board :**_doBottom_(_getBoard_(T))

- rotateLeft
  getScore(_rotateLeft_(T)) = _getScore_(T)
  needNext(_rotateLeft_(T)) = false
  isRunning(_rotateLeft_(T)) = true
  getBoard(_rotateLeft_(T)) = **Board :**_doRotateLeft_(_getBoard_(T))

- rotateRight
  getScore(_rotateRight_(T)) = _getScore_(T)
  needNext(_rotateRight_(T)) = false
  isRunning(_rotateRight_(T)) = true
  getBoard(_rotateRight_(T)) = **Board :**_doRotateRight_(_getBoard_(T))

- step
  Board :*isBottom*(*getBoard*(T)) && *needNext*(*getBoard*(T)) <=>
  getScore(*step*(T)) = *getScore*(T)
  Board :*isBottom*(*getBoard*(T)) && ¬*needNext*(*getBoard*(T)) <=>
  getScore(*step*(T)) = *getScore*(T) + 20 + **Board :***getNbLastCleaned*(*getBoard*(*step*(T)))*50
  Board :*isBottom*(*getBoard*(T)) && <=> *needNext*(*step*(T)) = false
  ¬**Board :***isBottom*(*getBoard*(T)) => *getScore*(*step*(T)) = *getScore*(T)
  ¬**Board :***isBottom*(*getBoard*(T)) <=> *getBoard*(*step*(T)) = **Board :***step*(*getBoard*(T))

- next
  getScore(*next*(T)) = *getScore*(T)
  needNext(*next*(T)) = false
  getBoard(*next*(T)) = **Board :***insert*(*getBoard*(), **Block :***init*(b))