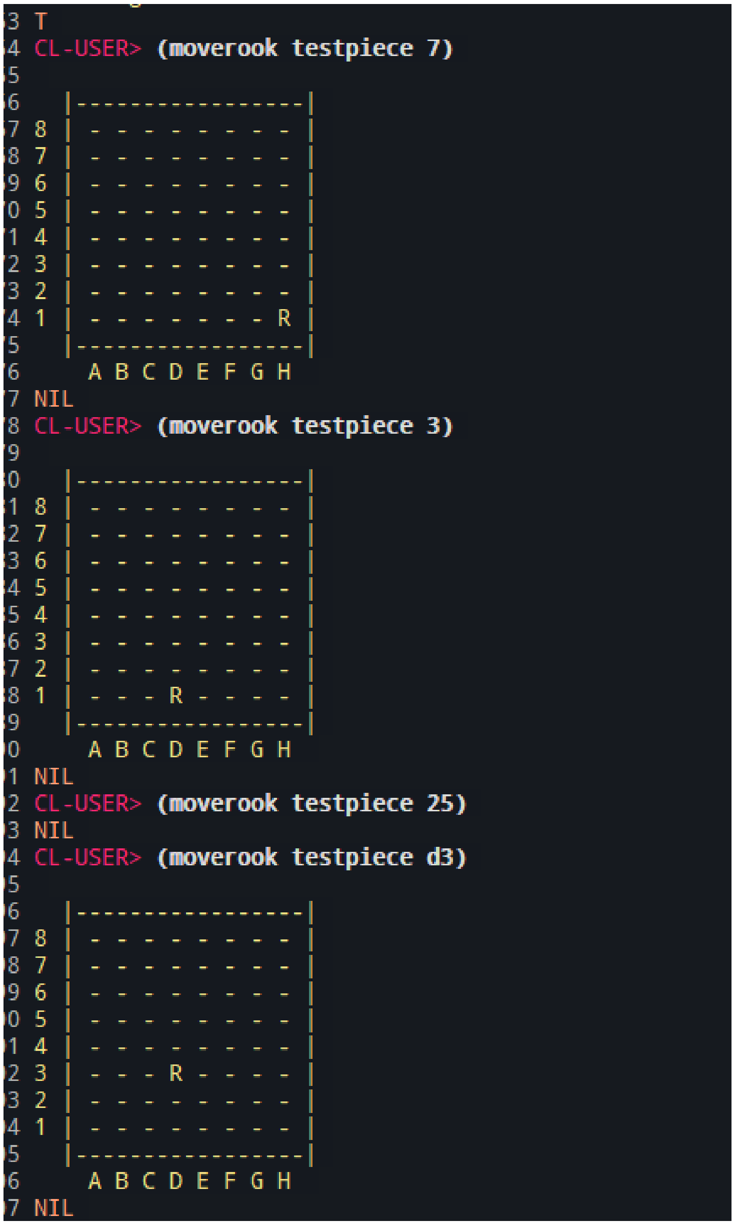
Chess AI Player Task Three: Rook Piece

Abstract: The goal of this task was to add the rook piece to the game and to be able to move it around the board and display its current position on the board. Orgainaly this task also included being able to move the piece around the board randomly. However after some trouble I have decided to make another task for that.

Demo:

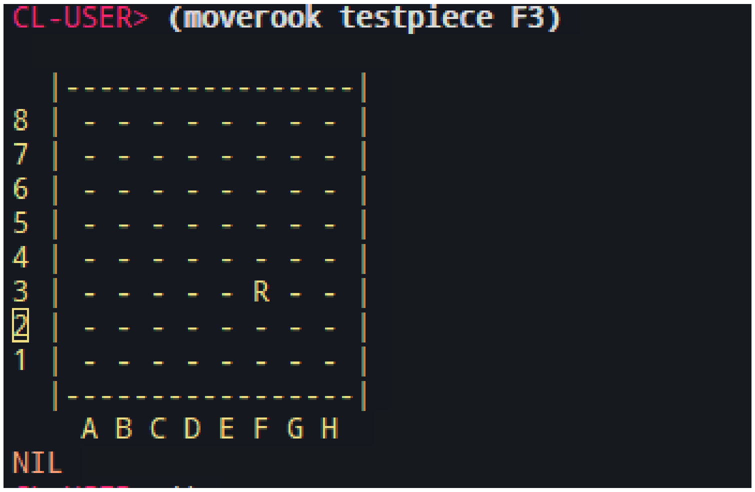


```
T
CL-USER> (moverook testpiece 7)

  |-----------------|
8 | - - - - - - - - |
7 | - - - - - - - - |
6 | - - - - - - - - |
5 | - - - - - - - - |
4 | - - - - - - - - |
3 | - - - - - - - - |
2 | - - - - - - - - |
1 | - - - - - - - R |
  |-----------------|
    A B C D E F G H
NIL
CL-USER> (moverook testpiece 3)

  |-----------------|
8 | - - - - - - - - |
7 | - - - - - - - - |
6 | - - - - - - - - |
5 | - - - - - - - - |
4 | - - - - - - - - |
3 | - - - - - - - - |
2 | - - - - - - - - |
1 | - - - R - - - - |
  |-----------------|
    A B C D E F G H
NIL
CL-USER> (moverook testpiece 25)
NIL
CL-USER> (moverook testpiece d3)

  |-----------------|
8 | - - - - - - - - |
7 | - - - - - - - - |
6 | - - - - - - - - |
5 | - - - - - - - - |
4 | - - - - - - - - |
3 | - - - R - - - - |
2 | - - - - - - - - |
1 | - - - - - - - - |
  |-----------------|
    A B C D E F G H
NIL
```



```
CL-USER> (moverook testpiece F3)

  |-----------------|
8 | - - - - - - - - |
7 | - - - - - - - - |
6 | - - - - - - - - |
5 | - - - - - - - - |
4 | - - - - - - - - |
3 | - - - - - R - - |
2 | - - - - - - - - |
1 | - - - - - - - - |
  |-----------------|
    A B C D E F G H
NIL
```

Code:

```
(defclass rook ()
  ((color
    :initform 'b
    :accessor rook-color)
   (current-square
    :initform 34
    :accessor current-square)
   (new-sqaure
    :initform nil
    :accessor desired-sqaure)))

(defun same-file (cs ns)
  (cond ((= (mod (- cs ns) 8) 0))))

(defun in-rank1 (square)
  (cond ((and ( >= square 0) (<= square 7)))))

(defun in-rank2 (square)
  (cond ((and ( >= square 8) (<= square 15)))))

(defun in-rank3 (square)
  (cond ((and ( >= square 16) (<= square 23)))))

(defun in-rank4 (square)
  (cond ((and ( >= square 24) (<= square 31)))))

(defun in-rank5 (square)
  (cond ((and ( >= square 32) (<= square 39)))))

(defun in-rank6 (square)
  (cond ((and ( >= square 40) (<= square 47)))))

(defun in-rank7 (square)
  (cond ((and ( >= square 48) (<= square 55)))))

(defun in-rank8 (square)
  (cond ((and ( >= square 56) (<= square 63)))))

(defun same-rank (cs ns)
  (cond
    ((and (in-rank1 cs) (in-rank1 ns)))
    ((and (in-rank2 cs) (in-rank2 ns)))
    ((and (in-rank3 cs) (in-rank3 ns)))
    ((and (in-rank4 cs) (in-rank4 ns)))
    ((and (in-rank5 cs) (in-rank5 ns)))
    ((and (in-rank6 cs) (in-rank6 ns)))
    ((and (in-rank7 cs) (in-rank7 ns)))
    ((and (in-rank8 cs) (in-rank8 ns)))
  )
)


(defun legal-rook-move (cs ns)
  (cond
    ((same-file cs ns))
    ((same-rank cs ns))
    )
  )

(setf testpiece (make-instance 'rook))
(setf (current-square testpiece) 63)
(get-square 63 2)

(defmethod moverook ((obj rook) (x integer))
  (setf (desired-sqaure obj) x)
  (cond
    ((legal-rook-move (current-square obj) (desired-sqaure obj))
     (setf (aref the-board (desired-sqaure obj)) 2)
     (setf (aref the-board (current-square obj)) 0)
     (setf (current-square obj) x)
     (display-current-board))
    )
  )
```