CheckerBoard(void) – [test_CheckerBoard_Constructor]

| Input: N/A | Out | out: N | N/A | | | | | |
|------------|---------------------------------|------------------------------|-------|------|---|---|---|---|
| State: N/A | State: | | | | | | | |
| | x: 12 o: 12 viab x: [S | 2 leDir SE, S NE, N | ectic | ons: | | | | |
| | х | * | x | * | х | * | х | * |
| | * | х | * | х | * | х | * | х |
| | х | * | х | * | х | * | х | * |
| | * | | * | | * | | * | |
| | | * | | * | | * | | * |
| | * | 0 | * | o | * | o | * | О |
| | 0 | * | 0 | * | o | * | 0 | * |
| | * | 0 | * | o | * | o | * | O |
| | | | | | | | | |

whatsAtPos(BoardPosition pos) – [test_whatsAtPos_MinRowMinCol]

| Input: (0,0) | Output: x |
|---|----------------------------------|
| State: pieceCount: | State: |
| x: 12 o: 12 | State of the object is unchanged |
| viableDirections: x: [SE, SW] o: [NE, NW] | |

| boar | d: | | | | | | |
|-------------------------|----------|-------|------|--------|------|--------|--------|
| х | * | х | * | х | * | х | * |
| * | х | * | х | * | х | * | х |
| х | * | х | * | х | * | х | * |
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | 0 | * | 0 | * | 0 | * | 0 |
| 0 | * | 0 | * | 0 | * | 0 | * |
| * | 0 | * | 0 | * | 0 | * | О |
| | | | | | | | |
| /hats | AtPo | os(Bo | ardP | ositic | n po | s) – [| test_v |
| Inpu | t: (7, | 7) | | | | | |
| State | е: | | | | | | |
| piece x: 12 o: 12 | <u> </u> | ınt: | | | | | |
| viabl x: [S o: [N | E, S | | ns: | | | | |

board:

Х

Х

Х

Χ

Χ

Х

*

Х

Х

Х

*

Х

Х

X

whatsAtPos(BoardPosition pos) – [test_whatsAtPos_MidRowMidCol]

| In | put: | (2. | 4 |
|----|------|-----|---|
| | | | |

State:

pieceCount:

x: 12 o: 12

viableDirections:

x: [SE, SW] o: [NE, NW]

board:

| х | * | х | * | х | * | х | * |
|---|--------|---|--------|---|--------|---|--------|
| * | х | * | х | * | х | * | х |
| х | * | х | * | х | * | х | * |
| * | | * | | * | | * | |
| | | | | | | | |
| | * | | * | | * | | * |
| * | * O | * | * O | * | * O | * | * O |
| * | | * | | * | | * | |

Output: x

State:

State of the object is unchanged

whatsAtPos(BoardPosition pos) – [test_whatsAtPos_BlackTile]

| Input | Input: (0, 1) | | | | | | Output: * | | | | |
|---------------------------|----------------|------------|----|---|---|---|-----------|----------------------------------|--|--|--|
| State | State: | | | | | | State: | | | | |
| piece x: 12 o: 12 | | nt: | | | | | | State of the object is unchanged | | | |
| viable x: [SI o: [N | E, SV E, N\ | V] | s: | | | | | | | | |
| х | * | х | * | х | * | х | * | | | | |
| * | х | * | х | * | х | * | х | | | | |
| х | * | х | * | х | * | х | * | | | | |
| * | | * | | * | | * | | | | | |
| | * | | * | | * | | * | | | | |
| * | 0 | * | 0 | * | 0 | * | 0 | | | | |
| o | * | 0 | * | 0 | * | 0 | * | | | | |
| * | О | * | 0 | * | 0 | * | 0 | | | | |

whatsAtPos(BoardPosition pos) - [test_whatsAtPos_EmptyTile

| Inpu | t: (4, | 0) | | | | | | Output: " " (empty space) |
|----------------|------------------|--------------|-------|--------|-------|-------|--------|--|
| State | е. | | | | | | | State: |
| | eCou | ınt: | | | | | | |
| x: 12 o: 12 | | | | | | | | State of the object is unchanged |
| ما ماند | la Dia | 4: | | | | | | |
| | iedire SE, SI | ectior W] | 15. | | | | | |
| o: [N | IE, N | W] | | | | | | |
| boar | d: | | | | | | | |
| | | | | | | | | |
| X | * | Х | * | Х | * | Х | * | |
| * | х | * | х | * | х | * | х | |
| x | * | х | * | х | * | x | * | |
| * | | * | | * | | * | | |
| | * | | * | | * | | * | |
| * | 0 | * | 0 | * | 0 | * | 0 | |
| 0 | * | 0 | * | 0 | * | 0 | * | |
| * | 0 | * | 0 | * | О | * | 0 | |
| | | | | | | | | |
| | D: | (D | | .,. | | | | |
| place | Piece | e(Ros | ardPo | Sition | n pos | , cha | r play | er) – [test_placePiece_EmptyTile_player] |

X]

| Input: pos = (3, 3) player = "x" | Output: N/A State: |
|---|---|
| State: pieceCount: x = 12 o = 12 | pieceCount: x = 13 o = 12 |
| viableDirections: x = [SE, SW] o = [NE, NW] | viableDirections: x = [SE, SW] o = [NE, NW] |

| * | х | * | х | * | Х | * |
|---|-----------|---|---|---------------------------------------|---|---|
| х | * | х | * | х | * | Х |
| * | х | * | х | * | х | * |
| | * | | * | | * | |
| * | | * | | * | | * |
| О | * | o | * | o | * | 0 |
| * | 0 | * | 0 | * | 0 | * |
| О | * | 0 | * | О | * | 0 |
| | x * * O * | X | x | X X X X X X X X X X X X X X X X X X X | X | X |

| x * | * X | x * | * X | X | * X |
|--------|-----|--------|--------|---------------------------------------|---------------------------------------|
| | | | х | * | Х |
| х | * | | | | |
| 1 | | Х | * | х | * |
| * | х | * | | * | |
| | * | | * | | * |
| * | o | * | 0 | * | 0 |
| 0 | * | 0 | * | 0 | * |
| * | 0 | * | 0 | * | 0 |
| | * | * * O | * | * * * * * * * * * * * * * * * * * * * | * * * * * * * * * * * * * * * * * * * |

placePiece(BoardPosition pos, char player) – [test_placePiece_cornerTile_replaceO_withX]

Input: pos = (7, 7)player = "o" State: pieceCount: x = 12o = 12 viableDirections: x = [SE, SW]o = [NE, NW]Χ Χ Χ Χ Χ Χ Χ Χ Х Χ Χ

0

0

0

0

0

0

0

0

Output: N/A

State:

pieceCount:

x = 13

o = 11

viableDirections:

x = [SE, SW]

o = [NE, NW]

| $\overline{}$ | <u>, , , , , , , , , , , , , , , , , , , </u> | | | | | | |
|---------------|---|---|---|---|---|---|---|
| х | * | х | * | х | * | х | * |
| * | х | * | х | * | х | * | х |
| х | * | х | * | х | * | х | * |
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | 0 | * | 0 | * | 0 | * | 0 |
| o | * | 0 | * | 0 | * | 0 | * |
| * | o | * | 0 | * | 0 | * | х |
| | | | | | | | |

| * | 0 | * | 0 | * | 0 | * | 0 | |
|---|---|---|---|---|---|---|---|--|
| | | | | | | | | |

placePiece(BoardPosition pos, char player) – [test_placePiece_WhiteTile]

Input: pos = (3,0) player = "x"

State: pieceCount: x = 12

o = 12

viableDirections: x = [SE, SW] o = [NE, NW]

Χ Χ Χ Χ Χ Х Χ Х Χ Х Χ Χ 0 0 0 0 0 0 0 0 0 0 0 0 Output: N/A

State:

pieceCount: x = 13

o = 12

viableDirections:

x = [SE, SW]o = [NE, NW]

| х | * | х | * | х | * | х | * |
|---|---|---|---|---|---|---|---|
| * | х | * | х | * | х | * | х |
| х | * | х | * | х | * | х | * |
| х | | * | | * | | * | |
| | * | | * | | * | | * |
| * | o | * | 0 | * | o | * | 0 |
| 0 | * | 0 | * | 0 | * | 0 | * |
| * | o | * | 0 | * | o | * | 0 |
| | | | | | | | |

placePiece(BoardPosition pos, char player) – [test_placePiece_middleTile_replaceX_withO]

| Inpu pos playe | t: = (2,4 er = " | 1) o" | | | | | | Out | put: I | N/A | | | | | |
|-----------------------------------|------------------------|---------------|----------|-----|----------|-----|----------|---------------------------|-------------|--------|---|----------|----------|----------|---|
| x = 1 o = 2 viable x = [| eCou 12 | ectior SW] | ns: | | | | | x = o = viat x = | | rectio | | | | | |
| | * | <u> </u> | * | T., | * | T., | * | х | * | х | * | х | * | х | * |
| X | <u> </u> | X | | X | ļ | X | <u> </u> | * | x | * | х | * | х | * | х |
| * | Х | * | Х | * | Х | * | Х | x | * | х | * | 0 | * | х | * |
| х | * | х | * | х | * | х | * | * | | * | | * | | * | |
| * | | * | | * | | * | | | * | | * | | * | | * |
| | * | | * | | * | | * | | <u> </u> ^_ | | | | <u> </u> | | |
| * | 0 | * | 0 | * | 0 | * | 0 | * | 0 | * | 0 | * | 0 | * | 0 |
| - | * | | * | 1_ | * | - | * | О | * | 0 | * | 0 | * | 0 | * |
| 0 | <u> </u> | 0 | <u> </u> | 0 | <u> </u> | 0 | | * | 0 | * | 0 | * | 0 | * | 0 |
| * | 0 | * | 0 | * | 0 | * | 0 | | | | | <u> </u> | <u> </u> | <u> </u> | |
| | | | | | | | | | | | | | | | |

placePiece(BoardPosition pos, char player) – [test_placePiece_CornerTile_replacex_withX]

| Input: | Output: N/A |
|-------------------|-------------------|
| pos = (0, 0) | |
| player = "X" | State: |
| | pieceCount: |
| State: | x = 12 |
| pieceCount: | o = 12 |
| x = 12 | |
| o = 12 | viableDirections: |
| | x = [SE, SW] |
| viableDlrections: | o = [NE, NW] |
| x = [SE, SW] | |
| o = [NE, NW] | |

| * | х | * | х | * | Х | * |
|---|-----------|---|---|---|---|---|
| х | * | х | * | х | * | х |
| * | x | * | х | * | х | * |
| | * | | * | | * | |
| * | | * | | * | | * |
| О | * | 0 | * | 0 | * | 0 |
| * | 0 | * | 0 | * | 0 | * |
| О | * | 0 | * | o | * | o |
| | x * * O * | X | x | X | X | X |

| Х | * | х | * | х | * | х | * |
|---|---|---|---|---|---|---|---|
| * | х | * | х | * | х | * | х |
| х | * | х | * | х | * | х | * |
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | 0 | * | 0 | * | 0 | * | 0 |
| 0 | * | o | * | 0 | * | 0 | * |
| * | o | * | 0 | * | o | * | 0 |

getPieceCounts(void) - [test_getPieceCounts_x12_o12]

Input: N/A

State:

pieceCount:

x: 12 o: 12

viableDirections:

x: [SE, SW] o: [NE, NW]

board:

| х | * | х | * | х | * | Х | * |
|---|---|---|---|---|---|---|---|
| * | х | * | х | * | х | * | х |
| х | * | х | * | х | * | х | * |
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | 0 | * | 0 | * | 0 | * | 0 |
| 0 | * | 0 | * | 0 | * | 0 | * |

Output:

x: 12 o: 12

State:

State of the object is unchanged

getViableDirections(void) - [test_getViableDirections_8x8board]

Input: N/A

State:

pieceCount:

x = 12

o = 12

viableDirections:

x = [SE, SW]

o = [NE, NW]

| х | * | х | * | х | * | х | * |
|---|---|---|---|---|---|---|---|
| * | х | * | Х | * | Х | * | Х |
| х | * | х | * | х | * | х | * |
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | 0 | * | 0 | * | 0 | * | 0 |
| o | * | 0 | * | 0 | * | 0 | * |
| * | 0 | * | 0 | * | 0 | * | 0 |

Output:

HashMap<Character, ArrayList<DirectionEnum>>

ViableDirections: x = [SE, SW], o =

[NE, NW]

State:

State of the pieceCount is unchanged
State of the board in unchanged
State of the HashMap is unchanged

checkPlayerWin(Character player) - [test_checkPlayerWin_no_opponent_pieces_left]

| Input: x | | | | | | | Output: True |
|---|------|-----|---|---|---|---|---|
| State: | | | | | | | State: State of the object is unchanged |
| pieceCou x: 12 o: 0 | ınt: | | | | | | unonangea |
| viableDire x: [SE, S' o: [NE, N board: | W] | ns: | | | | | |
| x * | х | * | х | * | х | * | |
| * x | * | х | * | х | * | х | |
| x * | х | * | х | * | х | * | |
| * | * | | * | | * | | |
| * | | * | | * | | * | |
| * | * | | * | | * | | |
| * | | * | | * | | * | |
| * | * | | * | | * | | |
| | | | | | | | |

checkPlayerWin(Character player) – [test_checkPlayerWin_opponent_pieces_exist]

| Input: x | Output: False |
|--|---|
| State: pieceCount: x: 12 o: 12 | State: State of the object is unchanged |
| viableDirections: x: [SE, SW] o: [NE, NW] board: | |

| | х | * | х | * | х | * | х | * |
|---|---|---|---|---|---|---|---|---|
| ĺ | * | х | * | х | * | х | * | х |
| | Х | * | х | * | х | * | х | * |
| | * | | * | | * | | * | |
| | | * | | * | | * | | * |
| | * | 0 | * | 0 | * | 0 | * | 0 |
| | 0 | * | 0 | * | 0 | * | 0 | * |
| | * | 0 | * | 0 | * | 0 | * | 0 |
| | | | | | | | | |

crownPiece(BoardPosition posOfPlayer) - [test_crownPiece_playerx_on_opposing_side]

Input: (7,1)

State:
pieceCount:
x: 12

Output: posOfPlayer = [X]

State:
pieceCount:
x: 12

viableDirections:viableDirections:x: [SE, SW]x: [SE, SW, NE, NW]o: [NE, NW]o: [NE, NW]

board:

| boar | <u>d:</u> | | | | | | |
|------|-----------|---|---|---|---|---|---|
| х | * | x | * | х | * | х | * |
| * | х | * | х | * | х | * | X |
| x | * | | * | х | * | Х | * |
| * | | * | | * | | * | |
| | * | | * | 0 | * | 0 | * |
| * | | * | 0 | * | 0 | * | 0 |
| | * | | * | 0 | * | 0 | * |
| * | х | * | 0 | * | 0 | * | 0 |
| | | | | | | | |

board:

| | ٠. | | | | | | |
|---|----|---|---|---|---|---|---|
| х | * | х | * | х | * | X | * |
| * | х | * | х | * | х | * | х |
| х | * | | * | х | * | х | * |
| * | | * | | * | | * | |
| | * | | * | 0 | * | 0 | * |
| * | | * | 0 | * | 0 | * | 0 |
| | * | | * | 0 | * | 0 | * |
| * | Х | * | 0 | * | 0 | * | 0 |
| | | | | | | | |

crownPiece(BoardPosition posOfPlayer) - [test_crownPiece_playerx_on_player_side]

| Inpu | t: (0,0 | 0) | | | | | | Output: N/A |
|----------------------------------|--------------------------|------------|-----|---|---|---|---|--|
| State piece x: 12 o: 12 | eCou | nt: | | | | | | State: State of the object is unchanged |
| x: [S | eDire E, S\ IE, N\ | V] | is: | | | | | |
| х | * | х | * | х | * | х | * | |
| * | х | * | х | * | х | * | х | |
| х | * | х | * | х | * | х | * | |
| * | | * | | * | | * | | |
| | * | | * | | * | | * | |
| * | 0 | * | 0 | * | 0 | * | 0 | |
| О | * | 0 | * | 0 | * | 0 | * | |
| * | 0 | * | 0 | * | 0 | * | 0 | |
| | - | | | | - | - | | |

crownPiece(BoardPosition posOfPlayer) – [test_crownPiece_playerx_already_crowned]

| Input: (4,0) | Output: N/A |
|---|---|
| State: pieceCount: x: 12 o: 12 | State: State of the object is unchanged |
| viableDirections: x: [SE, SW, NE, NW] o: [NE, NW] | |
| board: | |

| х | * | х | * | х | * | х | * |
|---|---|---|---|---|---|---|---|
| * | х | * | х | * | х | * | х |
| | * | х | * | х | * | х | * |
| * | | * | | * | | * | |
| Х | * | | * | | * | 0 | * |
| * | 0 | * | 0 | * | 0 | * | |
| 0 | * | О | * | 0 | * | 0 | * |
| * | 0 | * | 0 | * | 0 | * | 0 |
| | | | | | - | | |

movePiece(BoardPosition startingPos, DirectionEnum dir) – [test_movePiece_in_empty_spot]

Input: ((2,0), SE)

State:

pieceCount:

x: 12 o: 12

viableDirections:

x: [SE, SW] o: [NE, NW]

board:

| Dogr | | | | | | | |
|------|---|---|---|---|---|---|---|
| х | * | x | * | x | * | X | * |
| * | х | * | х | * | х | * | Х |
| х | * | х | * | х | * | Х | * |
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | o | * | 0 | * | 0 | * | 0 |
| О | * | 0 | * | 0 | * | 0 | * |
| * | o | * | 0 | * | 0 | * | 0 |

Output: BoardPosition object = (3,1)

State:

startingPos = null

pieceCount:

x: 12 o: 12

viableDirections:

x: [SE, SW] o: [NE, NW]

board:

| х | * | х | * | х | * | х | * |
|---|---|---|---|---|---|---|---|
| * | х | * | х | * | х | * | х |
| | * | х | * | х | * | х | * |
| * | х | * | | * | | * | |
| | * | | * | | * | | * |
| * | 0 | * | 0 | * | 0 | * | 0 |
| О | * | o | * | О | * | 0 | * |

| | * | 0 | * | 0 | * | 0 | * | 0 | |
|--|---|---|---|---|---|---|---|---|--|
| | | | | | | | | | |

movePiece(BoardPosition startingPos, DirectionEnum dir) – [test_movePiece_in_occupied_spot]

Input: ((3,1), SW)

State:

pieceCount:

x: 12

o: 12

viableDirections:

x: [SE, SW]

o: [NE, NW]

board:

| х | * | Х | * | Х | * | х | * |
|---|---|---|---|---|---|---|---|
| * | Х | * | Х | * | Х | * | Х |
| | * | х | * | х | * | х | * |
| * | Х | * | | * | | * | |
| o | * | | * | | * | | * |
| * | | * | 0 | * | 0 | * | 0 |
| o | * | 0 | * | 0 | * | 0 | * |
| * | o | * | o | * | o | * | 0 |

Output: N/A

State:

State of the object is unchanged

movePiece(BoardPosition startingPos, DirectionEnum dir) – [test_movePiece_SW_out_of_bounds]

Input: ((2,0), SW) Output: N/A State: State: pieceCount: State of the object is unchanged x: 12 o: 12 viableDirections: x: [SE, SW] o: [NE, NW] board: Χ Χ Χ Χ Х Χ Χ Χ Х Х Χ Х 0 0 0 0 0 0 0 0 0 0 0 0

jumpPiece(BoardPosition startingPos, DirectionEnum dir) - [test_jumpPiece_SEjump]

Output: Input: startingPos = new BoardPosition(4,4) x jumps over to (2,6) dir = DirectionEnum.SE o is removed True State: viableDirections: State: x: [SE, SW] viableDirections: o: [NE, NW] x: [SE, SW] o: [NE, NW] x is at (5,5) 0 is at (6,4) pieceCount x: 1 o: 0 pieceCount x: 1 o: 1

| boar | d: | | | | | | | boar | d: | | | | | | |
|------|----|---|---|---|---|---|---|------|----|---|---|---|---|---|---|
| | * | | * | | * | | * | | * | | * | | * | | * |
| * | | * | | * | | * | | * | | * | | * | | * | |
| | * | | * | | * | | * | | * | | * | | * | х | * |
| * | | * | | * | 0 | * | | * | | * | | * | | * | |
| | * | | * | х | * | | * | | * | | * | | * | | * |
| * | | * | | * | | * | | * | | * | | * | | * | |
| | * | | * | | * | | * | | * | | * | | * | | * |
| * | | * | | * | | * | | * | | * | | * | | * | |
| | | | | | | | | | | | | | | | |

jumpPiece(BoardPosition startingPos, DirectionEnum dir) – [test_jumpPiece_SWjump]

| Input: startingP dir = Dire | | | | Posit | tion(2 | 2,2) | Output: x jumps over to (4,0) o is removed True | | | | | | | | | |
|-----------------------------------|-----|----|---|-------|--------|------|--|-------|--------|-------|---|---|---|---|--|--|
| State: | 4: | | | | | | 04- | | | | | | | | | |
| viableDir | | S: | | | | | Sta | | rectio | ne. | | | | | | |
| o: [NE, N | - | | | | | | | SE, S | | ,,,,, | | | | | | |
| | - | | | | | | | ۷E, ۱ | | | | | | | | |
| pieceCou | ınt | | | | | | nicacCount | | | | | | | | | |
| x: 1 o: 1 | | | | | | | pieceCount x: 1 | | | | | | | | | |
| 0. 1 | | | | | | | 0: 0 | | | | | | | | | |
| x is at (2, | - | | | | | | | | | | | | | | | |
| o is at (3 | 1) | | | | | | boa | rd: | 1 | ı | ı | 1 | I | | | |
| board: | | | | | | | | * | | * | | * | | * | | |
| * | | * | | * | | * | * | | * | | * | | * | | | |
| * | * | | * | | * | | | * | | * | | * | | * | | |
| * | х | * | | * | | * | * | | * | | * | | * | | | |
| * O | * | | * | | * | | x | * | | * | | * | | * | | |

| | * | | * | | * | | * | * | | * | | * | | * | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| * | | * | | * | | * | | | * | | * | | * | | * | |
| | * | | * | | * | | * | * | | * | | * | | * | | |
| * | | * | | * | | * | | | | | | | | | | • |
| | | | | | | | | | | | | | | | | |

jumpPiece(BoardPosition startingPos, DirectionEnum dir) – [test_jumpPiece_invalidMove_noPieceToJump]

Input:

startingPos = new BoardPosition(3,3) dir = DirectionEnum.SE

State:

viableDirections:

x: [SE, SW]

o: [NE, NW]

pieceCount

x: 1

o: 1

x is at (3,3)

no piece to jump in SE direction

board:

| | * | | * | | * | | * |
|---|---|---|---|---|---|---|---|
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | | * | Х | * | | * | |
| | * | | * | | * | | * |
| * | | * | | * | | * | |
| | * | | * | | * | | * |
| * | | * | | * | | * | |

Output:

no move is made error and asked to retry

State:

viableDirections:

x: [SE, SW]

o: [NE, NW]

pieceCount

x: 1

o: 1

State of the board is unchanged

scanSurroundingPositions(BoardPosition startingPos) – [test_scanSurroundingPositions_validMoves]

| Input starti State x at (o (1, | ingPc e: (2,2) | os = r | new E | 3oard | Posit | tion(3 | Output: (0,0) " " (2,0) " " (0,2) " " | |
|---|----------------------|--------|-------|-------|-------|--------|---------------------------------------|---------------------------------|
| boar | d: | | | | | | | |
| | * | | * | | * | | * | State: |
| * | 0 | * | | * | | * | | State of the board is unchanged |
| | * | х | * | | * | | * | |
| * | | * | | * | | * | | |
| | * | | * | | * | | * | |
| * | | * | | * | | * | | |
| | * | | * | | * | | * | |
| * * * * | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |

scanSurroundingPositions(BoardPosition startingPos) – [test_scanSurroundingPositions_oSurroundings]

| Input: startingPos = new BoardPosition(4,4) | Output: (3,3) | | |
|--|---------------------------------|--|--|
| State: pieceCount: x: 3 | (5,3) (3,5) (5,5) " " | | |
| o: 1 | can move NW, SE | | |
| viableDirections | State: | | |
| x: [SE,SW] o: [NE, NW] | State of the board is unchanged | | |
| board: | | | |
| * * * * | | | |

| | * | | * | | * | | * | |
|---|---|---|---|---|---|---|---|---|
| Ì | | * | | * | | * | | * |
| ĺ | * | | * | х | * | Х | * | |
| j | | * | | * | 0 | * | | * |
| I | * | | * | Х | * | | * | |
| Ĭ | | * | | * | | * | | * |
| Ì | * | | * | | * | | * | |
| • | | | | | | | | |

scanSurroundingPositions(BoardPosition startingPos) – [test_scanSurroundingPositions_noValidMove]

| _ | | | | | | | | | |
|----|--|---|---|---|---|---|---|--|--|
| | Input: startingPos = new BoardPosition(0,0) State: pieceCount: x: 1 o: 0 viableDirections x: [SE,SW] o: [NE, NW] | | | | | | | Output: (1,1) " " (0,3) " " (2,0) " " no valid move directions State: State of the board is unchanged | |
| ١, | board: | | | | | | | | |
| | х | * | | * | | * | | * | |
| | * | | * | | * | | * | | |
| | * * * * | | | | | | | | |
| | * * * * | | | | | | | | |
| | * * * * | | | | | | | | |
| | * | | * | | * | | * | | |
| Н | | | | | | | | - | |

| getDirection(DirectionEnum dir) – [test_get | Direction_invalidDirection] |
|---|----------------------------------|
| Input: dir = DirectionEnum.NE | Output: |
| State: | direction: (-1,1) |
| State of the board doesn't affect the | State: |
| function | State of the object is unchanged |
| | |
| | |

What tests did each team member write? Just tell me the names of the functions (unless for some reason multiple team members wrote functions for the same method. In that case, tell me which tests specifically by giving me the test names)

| Laura | CheckerBoard(int) whatsAtPos(BoardPosition) getPieceCounts(void) |
|---------|---|
| Nadia | checkPlayerWin(Character) crownPiece(BoardPosition) movePiece(BoardPosition, DirectionEnum) |
| Keerthi | jumpPiece(BoardPosition, DirectionEnum) scanSurroundingPositions(BoardPosition) getDirection(DirectionEnum) |
| Meher | placePiece(BoardPosition, char) getViableDirections(void) |