Test Cases for Board Class

Required Test Cases

distributeStones

public int distributeStones(int startingPoint) throws PitNotFoundException

Helper method that distributes stones into pits and stores, skipping the opponent's store.

- Parameters: startingPoint - The starting pit
- Returns: The total number of stones added to pits and stores
- Throws: PitNotFoundException - If the pit number is invalid

Category of Test	Test Case	Expected Result
testDistributeStones	board.distributeStones(startingPoint)	4

getNumStones

public int getNumStones(int pitNum) throws PitNotFoundException

Gets the number of stones in a specific pit.

- Parameters: pitNum - The pit number
- Returns: The number of stones in the pit
- Throws: PitNotFoundException - If the pit number is invalid

Category of Test	Test Case	Expected Result
testGetNumStones	board.getNumStones(pitNum)	4

isSideEmpty

public boolean isSideEmpty(int pitNum) throws PitNotFoundException

Indicates whether one side of the board is empty. An empty side indicates the end of the game. more context: pits 1-6 are on one side of the board while pits 7-12 are on the other side of the board. if this method were called with a 3 as a parameter, it would return true if pits 1-6 were empty, false otherwise.

- Parameters: pitNum - The pit number
- Returns: true if the side of the board that includes the parameter pit number is empty
- Throws: PitNotFoundException

Category of Test	Test Case	Expected Result
testIsSideEmpty	board.isSideEmpty(pitNum)	False

registerPlayers

public void registerPlayers(Player one, Player two)

Connects Players to their Stores. Will need to call methods in store and in player to ensure a two-way connection

- Parameters: one - Player one, two - Player two

Category of Test	Test Case	Expected Result
testRegisterPlayers	registerPlayers(player1, player2)	assertEquals(player1, board.getStores().get(0).getOwner()); assertEquals(player2, board.getStores().get(1).getOwner());

Extra Test Cases

getStores

public ArrayList<Store> getStores()

Returns the list of stores in the board, not including the stores

- Returns: ArrayList of Stores

Category of Test	Test Case	Expected Result
testGetStores	getStores()	2

getPits

public ArrayList<Pit> getPits()

Returns the list of pits in the board, not including the stores

- Returns: ArrayList of pits

Category of Test	Test Case	Expected Result
testGetPits	getPits()	12

setUpPits

public void setUpPits()

Establishes 12 empty Pits in the board

Category of Test	Test Case	Expected Result
testSetUpPits	setUpPits()	12

initializeBoard

public void initializeBoard()

Initializes the board by distributing stones.

Category of Test	Test Case	Expected Result
testInitializeBoard	initializeBoard()	48

setUpStores

public void setUpStores()

Establishes 2 empty Stores in the board

Category of Test	Test Case	Expected Result
testSetUpStores	setUpStores()	2