resetBoard()

Description of Test	Test Case	Expected Result
Don’t initialize pits and stores	Call resetBoard alone	The pits and store are null and can not be reset
Initialize pits and stores	call initializeBoard first	The pits and stores are not null and therefore are reset properly

registerPlayers()

Description of Test	Test Case	Expected Result
Normal operation	Initialize stores and players	Both players have a not null store
Empty Stores	Don't initialize stores	IndexOutOfBoundsException
Null Player	Send a null player	NullPointerException

isSideEmpty()

Description of Test	Test Case	Expected Result
Normal Operation	Initialize call methods	False is returned as sides are not empty
One side not empty	Initialize and make one side not empty but the other empty	See that one side asserts false and the other asserts true
One pit not emptied	Empty 0-4 and leave pit 5	False as side is not empty
Invalid pit	Send -1 and 12	PitNotFoundException thrown

getNumStones()

Description of Test	Test Case	Expected Result
Normal Operation	Initialize board and pits	Correct assertions of 4
After Stone Modification	Initialize board and pits then modify a pit stone count	Correct assertion off 5
Invalid pit	Send -1 and 12	PitNotFoundException thrown

distributeStones()

Description of Test	Test Case	Expected Result
Normal Operation	Distribute from pit 1	Pits 2,3,4,5 will now have 5 stones
Distribute to correct store if crossed	Distribute from pit 6	Player 1 store should have value of 1, Player 2: 0, pits 7,8,9 should have 5
Distribute more than 4	Distribute from pit 6 with more than 4 stones (5)	Player 1 Store + 1, Pits 7,8,9,10 = 5
Distribute more stones than areas on board	Distribute 14 stones from pit 0	Every other pit = 5, pit 2 = 6, pit 1 = 1, Player 1 store = 1
Ensure correct call to nested function	Distribute until last stone lands on empty pit	Both the last stone pit and opposing pit are empty and correct score is added to store
Invalid Pit	Pass pit -1 and 12	PitNotFoundException

captureStones()

Description of Test	Test Case	Expected Result
Normal Operation	Pass pit 0 which has 1 stone in it	Store += pit 0 and opposing pit stones
Invalid Stopping Points	Pass -1 and 12 as pits	PitNotFoundException
Empty Pit used as stoppingPoint (hypothetical as not possible to happen)	Pass a pit with 0 stones in it	Opposing pits stones are added to the store
Opposing pit is empty	Pass pit 0 whilst pit 11 (12) is empty	1 stone is added to the store