White Box Testing

Tester: Umar Farooqi

Class: *Game*

ID	Test	Description	Result
1.01	*handleSelection*	Check interaction of square selection methods so that the appropriate action occurs when something is clicked on (e.g clicking on an enemy ship to start an attack if they are within range)	Pass
1.02	*moveSelectedShip*	Check that a ship is moved to the selected square if the square is available	Pass
1.03	*deleteShip*	Check that a ship is removed correctly from the map	Pass
1.04	*isShipSelected*	Check that the method correctly determines when a player's ship is selected	Pass
1.05	*getSelectedShip*	Check that the correct ship object is returned by the method	Pass
1.06	*getMoveSquares*	Check that the right set of moves is returned for a selected ship	Pass
1.07	*getEnemyShipCount*	Check that the correct number of enemy ships are counted	Pass
1.08	*createNewShips*	Check that the correct number of ships is created and that the ship count for each team reflects this	Pass
1.09	*startNewTurn*	Check that a new turn is correctly initiated (e.g resetting the number of moves available for each ship). If the turn is the first of the game, initialising of the game should be carried out (e.g creating ships for each team	Pass
1.10	*changeTurnNum*	Increments the number of turns taken so far if number of turns is greater than 1	Pass
1.11	*hasPlayerLost*	Check that it is currently determined if the player has lost based on if they still have any ships	Pass
1.12	*hasPlayerWon*	Check that the player has conquered all colleges	Pass

Class: *Map*

ID	Test	Description	Result
2.01	*isValidForEnemy*	Check if the square an enemy ship is moving to is within the boundaries of the map and unoccupied	Pass
2.02	*isValidSquare*	Check if the square selected by the player is pathable and unoccupied	Pass
2.03	*isNewSquare*	Check that the square the player selects to move to is not the square they already in	Pass
2.04	*getPossibleMoves*	Check that the set of moves available to the player in that turn is within the constraints of their movement	Pass

Class: *MapReader*

ID	Test	Description	Result
3.01	*getRawMapData()*	Check that a map can be read correctly from an appropriately formatted text file	Fail on first run - Solution was promptly found
3.02	*convertMap()*	Check that each map element is encoded into a Square object	Pass

Class: *Ship*

ID	Test	Description	Result
4.01	*nextToCoords*	Check that the method correctly identifies if a ship is next to the specified set of coordinates	Pass
4.02	*nextToShip*	Check that the method correctly identifies whether or not two given ships are next to each other	Pass

Class: *EnemyShipAI*

ID	Test	Description	Result
5.01	*isNearAShip*	Check it is correctly determined whether or not another ship is within the selected ship's range of possible moves	Pass
5.02	*moveShip*	Check that the ship is moved correctly as close to the nearest ship as possible	Pass
5.03	*randomMove*	Check that the randomly generated coordinates represent a valid move	Pass