def main (): while True: gameIsPlaying = True

turn = firstPlayer (randomly choose who plays first, player or computer) playerHitBoard = [‘ ‘] \* 10 #3x3 board and empty for user to hit compShipBoard compHitBoard = [‘ ‘] \* 10 #3x3 board and empty for comp to hit playerShipBoard print (Welcome message)

playerShipBoard = getPlayerBoard (user place 3x1 and 2x1 ships on 3x3 board, validate) compShipBoard = getCompBoard (use AI place 3x1 and 2x1 ships on 3x3 board correctly)

while gameIsPlaying: if turn == ‘player’: move = getPlayerHit (get players move to hit compShipBoard with O) playerHitBoard = checkHit (check player hit compShipBoard, X if hit or O)

displayBoard (print playerHitBoard)

print (hit/miss/sink)

gameIsPlaying = checkGameDone (check if someone won or board is full) turn = ‘computer’ else: move = getCompHit (get AI to hit playerShipBoard)

compHitBoard = checkHit (check comp hit playerShipBoard, X if hit or O)

displayBoard (print compHitBoard)

print (hit/miss/sink)

gameIsPlaying = checkGameDone (check if someone won or board is full) turn = ‘player’

if not playAgain (check if user wants to play again): print (Good Bye) break

Similarities to the Tic Tac Toe game is the board and the sequence of steps. Differences is that there are four boards instead of one, and the winning is if there is a hit, not four in a row. The program needs to store the boards of the game. Four boards are needed: playerHitBoard, compHitBoard, playerShipBoard, compShipBoard. They will be stored in lists.

**Test Cases:**

User enters ships overlapping User enters diagonal ships

User places ship off the board

User enters place already chosen

User hits

User sinks

User misses

User enters invalid data

User wants to play again

User wants to end game