L23: Pseudocode for public .doKnightsTour(), private .moveKnight(), and private .allTakenNearby()

http://en.wikipedia.org/wiki/Knight's_tour

<u>PRIVATE GLOBAL VALUES:</u>
- ChessBoard[ ][ ] is a 2D int array that stores all of the moves
- Horizontal[ ] is a 1D int array that stores the 8 different horizontal (col) moves
- Vertical[ ] is a 1D int array that stores the 8 different vertical (row) moves
- Taken[ ] is a 1D boolean array that tracks if the 8 squares around the knight are taken or not
- moveCount is an int that tracks how many moves have been made on the ChessBoard

```
//returns true if all locations around the knight are taken
//returns false if any locations around the knight are available
private boolean allTakenNearby()
{
        initialize local variables you need
        boolean allFilled = true;

        //reset the taken array to all positions being NOT taken
        For loop through TAKEN array positions from 1 to less than Taken array length
        {
                mark each location as **NOT TAKEN** in the array
        }

        //test each of the 8 possible locations around the knight one at a time
        For loop through TAKEN array positions from 1 to less than Taken array length
        {
                //DON'T CHANGE THE REAL GLOBAL myRow LOCATION OF THE KNIGHT ON THE BOARD!!!!
                //DON'T CHANGE THE REAL GLOBAL myCol LOCATION OF THE KNIGHT ON THE BOARD!!!!
                create a tempRow by adding the real myRow plus the Vertical[i] array value
                create a tempCol by adding the real myCol plus the Horizontal[i] array value


                if this temporary location is ON THE CHESSBOARD
                {
                        //we have moved to this location before, so it is **TAKEN**
                        if this temporary location ON THE CHESSBOARD has ALREADY BEEN VISITED
                        {
                                mark this location as TAKEN in the array
                        }
                        else
                        {
                                //at least one location around the knight is available!!
                                allFilled = false;
                        }
                }
                else    //this location is NOT ON THE CHESSBOARD, so it is ***TAKEN**
                {
                        mark this location as TAKEN in the array
                }
        }

        return allFilled;   //TRUE/FALSE, tells if all moves around the player are taken
}
```