Russell Miller 
ECS20, HW3
Chess Knight Configuration Problem

I wrote a program that calculates the number of valid configurations that 0 to 64 chess knights can be in, if no knight can capture another knight on a board. I was asked to prove why my program counts every configuration, but doesn't count an identical configuration twice.
I decided that I would try to explain my program using induction:

Kx,y = Knight at position (x, y) where (0,0) is the top left corner and (7,7) is the bottom right corner.
i will represent each location to be $K_i$, where i is the index of position 0 to 63.

**(Base case:**) The program starts at (0,0), moving row by row, up to (7,7), setting the positions to T or F.
**(Inductive step:)** For each $K_n$, $K_{n+1}$ is checked for if the position is valid depending on the current state of the board.
If the position is valid as T, then it is set to T.
It recursively does this for $K_{n+1}$ up to the final position (7,7).
At the final position (7,7), the ConfigurationCount is incremented.

The program then returns, until it reaches the most recent T value (which can be (7,7)), at which it then sets the position from T to F.
It then proceeds to recursively check again from that current position $K_n$, up until (7,7).
Each $K_{n+1}$ is part of a unique configuration because of the one change from T to F above (which is never reversed to F to T unless a previous value has been changed, making it a valid and unique configuration).

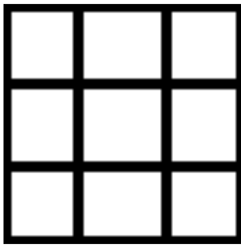
Every time the program reaches (7,7), it increments the ConfigurationCount, and repeats the process of setting the most recent T value to F, until all positions are set to F, at which point the program simply returns from the recursive function, returning the number of configurations.

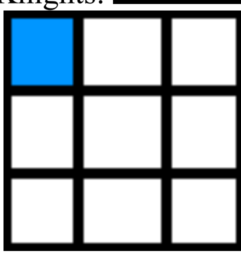
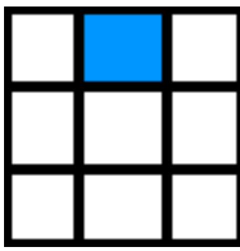
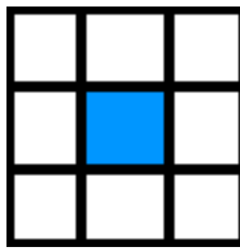
Side Note: You can't start by setting each position to F, and then setting the position to T afterward because of the conditions for which setting a position to T is valid. Since there are no special conditions for a position to be F, it can be considered the "default" value.

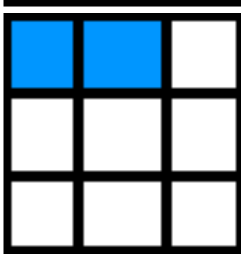
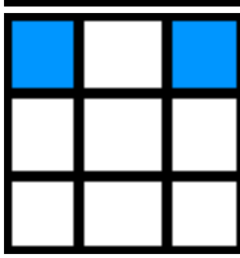
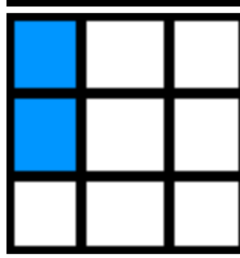
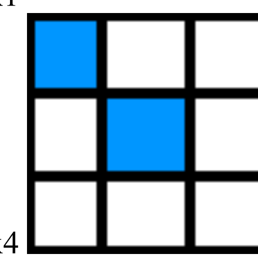
Russell Miller
ECS20, HW3
Chess Knight Configuration Problem

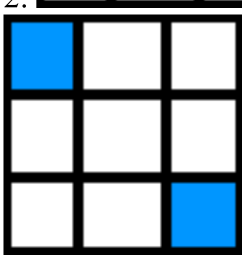
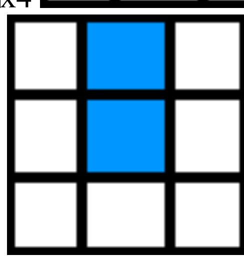
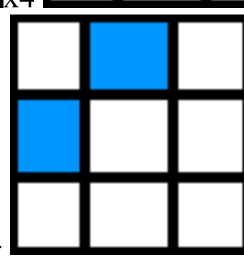
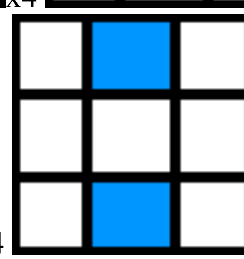
The following pictures show all the configurations for a 3x3 board. The pictures are the basic configurations that can each be rotated the number of times specified to create a unique configuration.

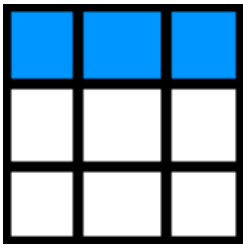
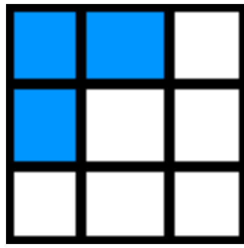
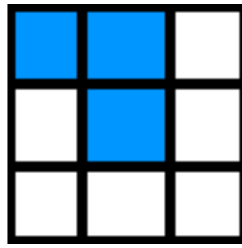
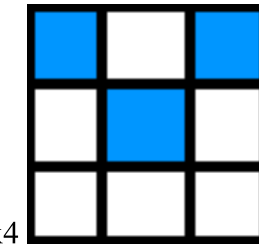
The total configurations for a 3x3 is 94 according to the below diagrams.
The program outputs
"Configurations: 94"
as an answer for a 3x3 board.

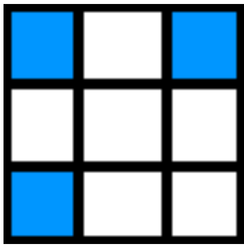
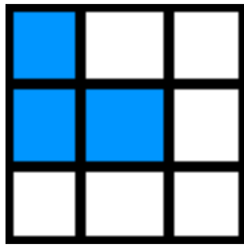
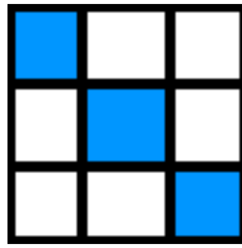
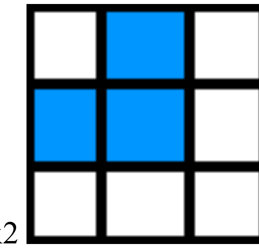
0 Knights:  x1

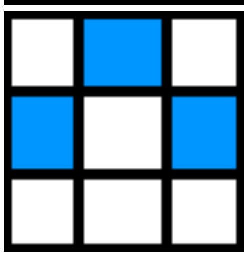
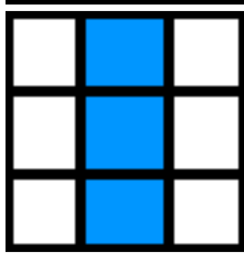
1:  x4  x4  x1

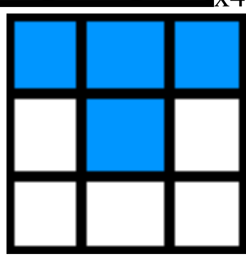
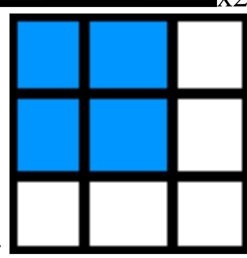
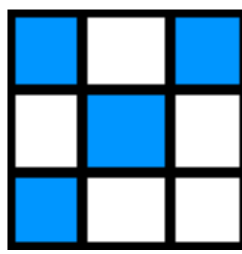
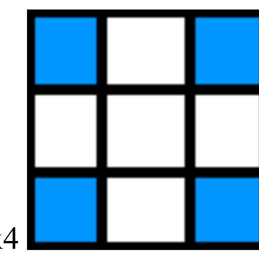
2:  x4  x4  x4  x4

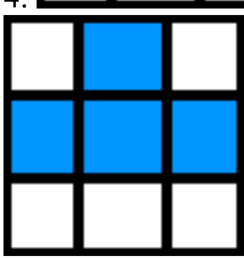
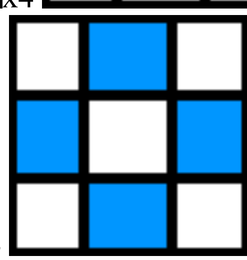
 x2  x4  x4  x2

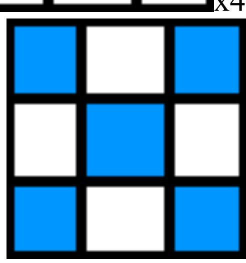
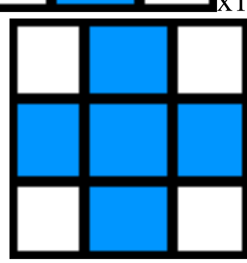
3:  x4  x4  x4  x4

 x4  x4  x2  x4

 x4  x2

4:  x4  x4  x4  x1

 x4  x1

5:  x1  x1