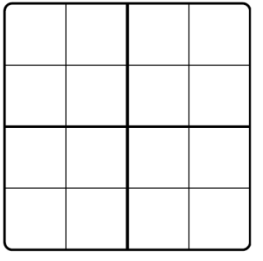
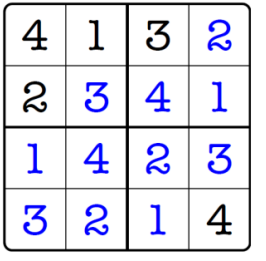
Task 1:

In a **2x2 Sudoku puzzle**, there are 4 connected 2x2 grids and the objective is to fill the grid with numbers from the set ***V*** = *{1, 2, 3, 4}*, such that each number appears exactly once in the same grid, exactly once in any row, and exactly once in any column. Below is a diagram depicting an empty 2x2 Sudoku and a possible solution state.





Write a program that takes an empty 2x2 grid as input and fills it to reach a valid solution state using backtracking.

Task 2:

Recall **N-Queen Problem**. Now assume that we alter the problem a bit and we now have to place **N+1** queens in an **NxN** grid.

Modify your N-Queen program to accommodate this new condition and then prove that this altered **N-Queen Problem** is now impossible to solve. Your program should also print the total number of "*backtrackings*" done during execution.