COM S 327, Spring 2018
Programming Project 0
The n Queens Problem

The n Queens Problem is a mathematical game in which the player attempts to place n queens on an $n \times n$ chessboard such that no two queens may attack each other. The game was invented about 170 years ago and has been studied by many mathematicians and computer scientists, including such luminaries as Carl Gauss.

You are to implement a solver for the 8 Queens Problem. The simplest solution will be recursive, though iterative solutions can work just as well.

Labeling the board columns a–h, left to right, and 1–8, bottom to top, your solutions should be printed 1 per line using 16 bytes, column row column row... Here is one solution to 8 Queens in the required format:

```
a2b4c6d8e3f1g7h5
```

In the game of chess, a queen can move (and attack) any unobstructed position in the same row, column, or diagonal from her starting position. Thus, all solutions to 8 Queens (there are 92 of them) have exactly one queen in each row, column, and diagonal of the chessboard.

A naïve solution will be very slow. That's okay; I don't care how fast your solution is. However, in a development cycle, testing a slow solution can be very painful. You may find it beneficial to start with implementations of 5, 6, or 7 queens (10, 4, and 40 solutions, respectively) solvers until you know your solution works correctly, then extend your board to 8×8.

See the syllabus for information about what to turn in and submission format. In particular, you must write, use, and turn in a working *Makefile*!