[Boggle](https://en.wikipedia.org/wiki/Boggle) is a word game that is played on a 4x4 board with 16 letter tiles. The goal is to find as many words as possible given a time constraint. For this exercise, we are making one modification. Now it is possible for one or more of the letter tiles to be blank (denoted by \*). When a tile is blank, it can be treated as any other letter. Note that in one game it does not have to be the same character for each word. For example, if the tiles C, T, and \* are adjacent. The words cot, cat, and cut can all be used. You will be given a text file containing all valid English words (a dictionary). The board will also be passed in as a text file with commas separating the letters. For example a file may contain:

A, C, E, D, L, U, G, \*, E, \*, H, T

This is equivalent to the board:

A C E D

L U G \*

E \* H T

Some sample words from this board are ace, dug, eight, hole, huge, hug, tide.

Choose one of the two challenges below to complete.

Challenge #1 (more fullstack):

Implement the user interface for a single person Boggle game. The player should be able to enter a word and then the program will validate that the word is in actually in the board and an English word. This is a very open-ended problem as it is meant to show how you think about the user and your frontend development skills.

Challenge #2 (more backend):

There are two main parts you have to implement for this challenge. When a user inputs a word, you need to verify whether it can be found in the Boggle board and if it is an English word. The second part is returning all possible words found in the board. We expect two functions. No UI is required, but we would like your code to be straightforward enough for us to run it to test both functions separately. If you want to do a text prompt, you may.