*4. Assignment #4 - Boggle\_Solver React App*

**About the Assignment**

For this project you will be creating a basic web app for playing Boggle in “solitaire” mode using the React library.

**Requirements**

1. Features [5 pts each]  
   -Display a running list of words found  
   -Notify users when they submit an answer they already found  
   -Start/Stop Button:  
   \*\* Hide the board until “Start” is clicked.  
   \*\* Display remaining words when “Stop” is clicked
2. Stretch Features [Extra Credit. 5 pts each - 20pts max]  
   -Timed game  
   -Make it pretty  
   -Generate random board  
   -Testing  
   -(anything you want)

***Boogie\_solver.js***

*/\*\**

*\* Given a Boggle board and a dictionary, returns a list of available words in*

*\* the dictionary present inside of the Boggle board.*

*\* @param {string[][]}* grid *- The Boggle game board.*

*\* @param {string[]}* dictionary *- The list of available words.*

*\* @returns {string[]} solutions - Possible solutions to the Boggle board.*

*\*/*

exports.findAllSolutions = function(grid, dictionary) {

let solutions = [];

return solutions;

}

const grid = [['T', 'W', 'Y', 'R'],

['E', 'N', 'P', 'H'],

['G', 'Z', 'Qu', 'R'],

['O', 'N', 'T', 'A']];

const dictionary = ['art', 'ego', 'gent', 'get', 'net', 'new', 'newt', 'prat',

'pry', 'qua', 'quart', 'quartz', 'rat', 'tar', 'tarp',

'ten', 'went', 'wet', 'arty', 'egg', 'not', 'quar'];

console.log(exports.findAllSolutions(grid, dictionary));