

Assignment 02 (Due: Wednesday, October 26, 2016, 11 : 59 : 00PM Central Time)

CSCE 322

THIS ASSIGNMENT IS ONLY WORTH 10% OF YOUR FINAL GRADE.

1 Instructions

In this assignment, you will be required to write JavaScript functions that simplify playing of the variation of **Greater Than Sudoku**.

1.1 Data File Specification

An example of properly formatted file is shown in Figure 1. The first file encodes a list of spaces to place values into; the second file encodes the values to be placed in the corresponding spaces, the third file encodes the visible grid; the fourth file encodes the inequalities between the rows (-1 means decreasing from top to bottom, 1 means increasing from top to bottom), and the fifth file encodes the inequalities between the columns (-1 means decreasing from left to right, 1 means increasing from left to right).

```

part01test01.spaces.gts
11,8,6,1,4,10,9,11,15,16,15

part01test01.values.gts
4,3,4,2,1,2,4,4,1,4,4

part01test01.game.gts
-,4,-,2
3,-,-,4
4,3,-,-
-,-,4,-

part01test01.vertical.gts
1,-1,-1,1
1,1,1,-1
-1,-1,1,1

part01test01.horizontal.gts
1,-1,-1
-1,-1,1
-1,-1,-1
-1,1,-1

```

Figure 1: A properly formatted game encoding

Spaces are defined from the upper-left corner of the game (Position 1), down the first column, down the second column... to the bottom of the last column.

2 One Player, One Move

The first part (`onePlayerOneMove` in the file `csce322hw02pt01.js`) will take in three (3) arguments (a game's seen values, signs between rows, and signs between columns) and return a function that takes in two arguments (a location and a value for a move), and returns the seen game that is the result of making the move in the initial game. If the space is occupied or the value would violate the relevant inequalities, the state of the game is unchanged.

3 One Player, Many Moves

The third part (`onePlayerManyMoves` in the file `csce322hw02pt03.js`) will take in three (3) arguments (a game's seen values, signs between rows, and signs between columns) and return a function that takes in two arguments (an array of locations and an array of values for moves), and returns the seen game that is the result of making the moves in the initial game. If a space is occupied or the value would violate the relevant inequalities, the state of the game is unchanged for that move and the next move (if there is one) is processed.

4 Naming Conventions

Your files should follow the naming convention of `csce322hw02pt01.js` and `csce322hw02pt02.js`.

4.1 `helpers.js`

A file named `helpers.js` has been provided with the functionality to read the `.gts` files into numerical matrices. If a modified `helpers.js` file is not included with your submission, the default will be used in its place.

5 webgrader Note

Submissions will be tested with `node.js`, note the browser. `cse.unl.edu` is currently running version 4.2.4 of `node`.

6 Point Allocation

Component	Points
<code>csce322hw02pt01.js</code>	50%
<code>csce322hw02pt02.js</code>	50%
Total	100%

7 External Resources

[JavaScript Tutorial](#)