Homework 1: Solving the Sudoku Problem

You will build a system to solve the Sudoku problem as a CSP based on consistency check and backtrack search. The objective of Sudoku is to fill a 9 × 9 grid with digits from 1 to 9 so that each column, each row, and each of the nine 3 × 3 sub-grids that compose the grid contain all of the digits from 1 to 9. The input is a partially completed grid and the output solution is a completely filled grid. For example, we have the following problem (a) and solution (b).

8			9	3				2
		9					4	
7		2	1			9	6	
2							9	
	6						7	
	7				6			5
	2	7			8	4		6
	3					5		
5				6	2			8

(a) Problem

8	4	6	9	3	7	1	5	2
3	1	9	6	2	5	8	4	7
7	5	2	1	8	4	9	6	3
2	8	5	7	1	3	6	9	4
4	6	3	8	5	9	2	7	1
9	7	1	2	4	6	3	8	5
1	2	7	5	9	8	4	3	6
6	3	8	4	7	1	5	2	9
5	9	4	3	6	2	7	1	8

(b) Solution

Input format: the input will be given to you in the form of a matrix with a "|" separating two numbers.
Output format: Your solution output should be a matrix as well with the missing values filled.

Input |8| | |9|3| | | |2|
| | | 9| | | | |4| |
…

Your output: |8|4|6|9|3|7|1|5|2|
|3|1|9|6|2|5|8|4|7|
…

Code submission requirements: (1) A **readme file** must accompany the **code** to describe (i) what each function is for, (ii) which functions are for consistency check and (iii) which functions are for backtrack search. (2) Your code should also have clear comments. If we cannot find your functions for consistency check and/or backtrack search, points will be deducted.

Programming language: You can use any language of your choice.

Demo date: Friday, 2/18/2022 from 9:00am – 12:00pm.

Deadline for code and readme file submission on Blackboard: 2/17/2022, 11:59pm (CST)