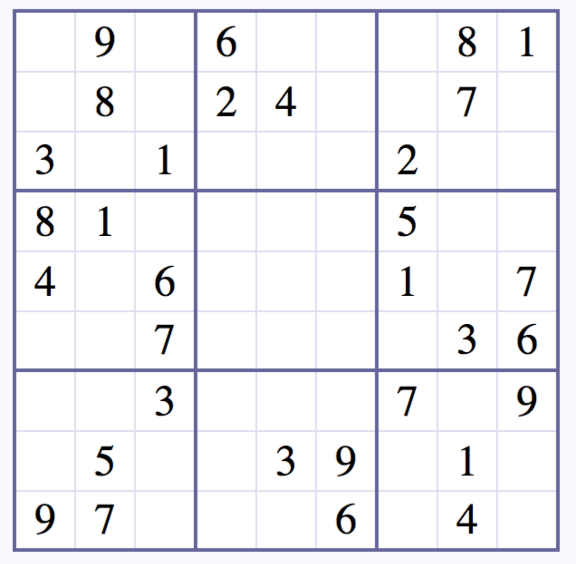
CMSC 671 Artificial Intelligence - Fall 2013
Homework Assignment 3
Due at the start of class on October 2^{nd}

From the Wikipedia Sudoku page:

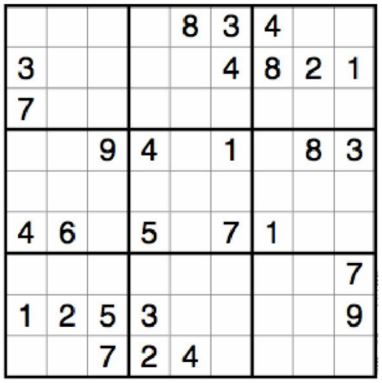
> "The objective is to fill a 9x9 grid with digits so that each column, each row, and each of the nine 3x3 sub-grids that compose the grid (also called "boxes", "blocks", "regions", or "sub-squares") contains all of the digits from 1 to 9. The puzzle setter provides a partially completed grid, which typically has a unique solution."

Below is an example of an easy Sudoku:



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

- Formulate solving a Sudoku puzzle as constraint satisfaction. Submit answers to the questions below. (20 points)
  - What are the variables?
  - What are their domains?
  - What are the constraints formulated as binary constraints?
  - Does the requirement that each digit has to occur in each row, column, and block have to be directly specified? Why or why not?
- Write a Sudoku solver that does not represent the requirement that each digit has to occur in each row, column, and block directly. Your solver must use the AC3 algorithm. (50 points)
  - The input to your algorithm should be a 9x9 grid of characters that are either digits or '*' if the digit for that location is unspecified. It's output should be in the same form. If no solution can be found then simply output "no solution possible".
  - Submit your code and the result of running your program on the Sudoku above.
- Consider the Sudoku below:



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

Look at the lower right block. For each cell in that block, say whether an 8 can be placed there and, if not, why not. What can we conclude about where the 8 can go and why? What general strategy does this suggest?

Note that arc consistency does not use this form of reasoning. Modify your solver to explicitly reason about the requirement that each digit has to occur in each row, column, and block directly. (30 points)

- Submit your code and the result of running your program on the Sudoku above. Also run your algorithm on the two Sudoku below and report the results (either a solution, the fact that no solution is possible, or the fact that your code never terminated).

If your code easily solves the two puzzles below, try this one:

```
**62*1***
8******71
**17***32
**7*3**4*
*5*****8*
*8**4*7**
46***58**
17******4
***4*65**
```

And this one:

```
***7****3
*96******
2**85****
17*2*4*36
*6**7**4*
*826*351*
****17**8
******25*
9****2***
```

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

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