CS 122 – Homework #5 – Sudoku – due Wednesday, March 20, 2019

The purpose of this assignment is to apply recursion to solve a puzzle. The puzzle to solve is called Sudoku, which is a popular Japanese number puzzle. The format of the puzzle is a 9 by 9 array of one-digit numbers, as in the following figure.

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

The object of the game is to fill in the blanks of the puzzle with the numbers 1-9 so that each number 1-9 appears exactly once in all 9 rows, all 9 columns, and all 9 square blocks of 3x3 cells.

Write a Java program that uses recursion with backtracking to solve this problem. Besides the main() method, there should be a solve() method that performs the recursion and backtracking. I recommend that you also write two more methods: an isGood() method that determines if a number placed in a certain cell obeys or violates the rules of Sudoku; and a printBoard() method that prints the entire 9x9 board.

You may assume that the input file will have the following format:
```
.6. 1.4 .5.
..8 3.5 6..
2.. ... ..1

8.. 4.7 ..6
..6 ... 3..
7.. 9.1 ..4

5.. ... ..2
..7 2.6 9..
.4. 5.8 .7.
```

The puzzle will have some predefined numbers in it. Do not change these values. A dot in the puzzle represents a space to be filled by your program. For readability, the input file also uses blanks and blank lines so that the 3x3 squares are easier to see. You will use this same format in the output.

Your program should be called Sudoku.java. The user will run your program from the command line using the command `java Sudoku <name of input file>`. The program should print all solutions to the puzzle. The output format should be the same as the input format shown above, with the periods replaced by real digits. Each solution should be introduced with a solution number message such as "Here is solution # 5". Also print two blank lines after each solution.

Good luck & enjoy!

For example, suppose a text file called input2.txt looks like this:

```
... ..4 .5.
..8 3.5 6..
2.. ... ..1

8.. 4.7 ..6
..6 ... 3..
7.. 9.1 ..4

5.. ... ..2
..7 2.6 9..
.4. 5.8 .7.
```

Then when we run "java Sudoku input2.txt" the output should be:

```
Here is solution # 1
963 174 258
178 325 649
254 689 731

821 437 596
496 852 317
735 961 824

589 713 462
317 246 985
642 598 173


Here is solution # 2
963 714 258
178 325 649
254 689 731

821 437 596
496 852 317
735 961 824

589 173 462
317 246 985
642 598 173
```