SudokuSolver.java

Background:

We worked together to create a Sudoku Generator (**SudokuGenerator.java**). Perhaps we could use this code, modifying it slightly, to create a Sudoku Solver (**SudokuSolver.java**).

Assignment:

Write the class **SudokuSolver.java**, and save it in your Sudoku directory. This program should open a text file, chosen by the user. You may want to make use of the **OpenFile** class. The contents of the text file will look something like this:

```
0 0 2 0 0 4 0 1 0
6 0 0 1 0 5 3 0 4
3 1 4 6 8 0 5 0 0
0 0 1 5 9 7 0 0 3
7 0 8 0 4 1 2 5 9
0 0 3 8 0 2 7 0 0
9 3 0 0 7 0 1 0 8
1 0 0 2 0 3 0 9 0
0 4 0 0 1 8 0 3 0
```

The numbers in the text file represent a Sudoku Puzzle, with some spots filled in with the numbers 1 through 9, and other spots "missing" a number, with a 0 as a place holder. Your program should solve the puzzle, filling in the "missing" numbers (represented by the 0's). Your program should print the progress of the puzzle (just as we did in **SudokuGenerator.java**), and the solution should be printed at the end.

See the following 2 sample run outputs.

```
C:\Java\Sudoku>java SudokuSolver a1.txt


row = 0,  col = 0,  index = 5,  arr[5] = 8
Array:    4  6  1  5  2  8  9  7  3
8 0 2 0 0 4 0 1 0
6 0 0 1 0 5 3 0 4
3 1 4 6 8 0 5 0 0
0 0 1 5 9 7 0 0 3
7 0 8 0 4 1 2 5 9
0 0 3 8 0 2 7 0 0
9 3 0 0 7 0 1 0 8
1 0 0 2 0 3 0 9 0
0 4 0 0 1 8 0 3 0
```

row = 0, col = 1, index = 3, arr[3] = 7
Array: 6 4 2 7 3 8 1 9 5
8 7 2 0 0 4 0 1 0
6 0 0 1 0 5 3 0 4
3 1 4 6 8 0 5 0 0
0 0 1 5 9 7 0 0 3
7 0 8 0 4 1 2 5 9
0 0 3 8 0 2 7 0 0
9 3 0 0 7 0 1 0 8
1 0 0 2 0 3 0 9 0
0 4 0 0 1 8 0 3 0

row = 0, col = 3, index = 1, arr[1] = 3
Array: 8 3 1 9 2 5 4 7 6
8 7 2 3 0 4 0 1 0
6 0 0 1 0 5 3 0 4
3 1 4 6 8 0 5 0 0
0 0 1 5 9 7 0 0 3
7 0 8 0 4 1 2 5 9
0 0 3 8 0 2 7 0 0
9 3 0 0 7 0 1 0 8
1 0 0 2 0 3 0 9 0
0 4 0 0 1 8 0 3 0

row = 0, col = 4, index = -1
Array: 7 8 4 6 3 2 9 1 5
8 7 2 3 0 4 0 1 0
6 0 0 1 0 5 3 0 4
3 1 4 6 8 0 5 0 0
0 0 1 5 9 7 0 0 3
7 0 8 0 4 1 2 5 9
0 0 3 8 0 2 7 0 0
9 3 0 0 7 0 1 0 8
1 0 0 2 0 3 0 9 0
0 4 0 0 1 8 0 3 0

MANY STEPS IN BETWEEN . . .

row = 8, col = 6, index = 8, arr[8] = 6
Array: 8 3 9 7 2 1 5 4 6
8 5 2 7 3 4 9 1 6
6 7 9 1 2 5 3 8 4
3 1 4 6 8 9 5 7 2
4 2 1 5 9 7 8 6 3
7 6 8 3 4 1 2 5 9
5 9 3 8 6 2 7 4 1
9 3 5 4 7 6 1 2 8
1 8 6 2 5 3 4 9 7
2 4 7 9 1 8 6 3 0

```
row = 8,  col = 8,  index = 3,  arr[3] = 5
Array:    9   6   1   5   2   7   8   3   4
8  5  2  7  3  4  9  1  6
6  7  9  1  2  5  3  8  4
3  1  4  6  8  9  5  7  2
4  2  1  5  9  7  8  6  3
7  6  8  3  4  1  2  5  9
5  9  3  8  6  2  7  4  1
9  3  5  4  7  6  1  2  8
1  8  6  2  5  3  4  9  7
2  4  7  9  1  8  6  3  5

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

C:\Java\Sudoku>java SudokuSolver a2.txt


row = 0,  col = 0,  index = 7,  arr[7] = 5
Array:    7   3   9   2   1   4   6   5   8
5  8  0  3  2  0  9  0  4
4  0  0  0  5  0  0  3  0
0  0  1  0  4  0  8  0  2
3  5  0  0  6  9  0  1  0
2  0  0  1  8  7  5  0  3
0  1  8  0  3  0  0  4  9
0  6  0  5  9  0  3  0  7
7  0  3  8  0  4  6  2  0
0  2  5  6  7  0  4  0  0

row = 0,  col = 2,  index = 6,  arr[6] = 6
Array:    2   5   1   9   8   4   6   7   3
5  8  6  3  2  0  9  0  4
4  0  0  0  5  0  0  3  0
0  0  1  0  4  0  8  0  2
3  5  0  0  6  9  0  1  0
2  0  0  1  8  7  5  0  3
0  1  8  0  3  0  0  4  9
0  6  0  5  9  0  3  0  7
7  0  3  8  0  4  6  2  0
0  2  5  6  7  0  4  0  0
```

row = 0, col = 5, index = 7, arr[7] = 1
Array: 9 4 5 7 8 3 6 1 2
5 8 6 3 2 1 9 0 4
4 0 0 0 5 0 0 3 0
0 0 1 0 4 0 8 0 2
3 5 0 0 6 9 0 1 0
2 0 0 1 8 7 5 0 3
0 1 8 0 3 0 0 4 9
0 6 0 5 9 0 3 0 7
7 0 3 8 0 4 6 2 0
0 2 5 6 7 0 4 0 0

MANY STEPS IN BETWEEN . . .

row = 8, col = 7, index = 1, arr[1] = 9
Array: 8 9 4 6 3 7 1 2 5
5 8 6 3 2 1 9 7 4
4 7 2 9 5 8 1 3 6
9 3 1 7 4 6 8 5 2
3 5 7 4 6 9 2 1 8
2 4 9 1 8 7 5 6 3
6 1 8 2 3 5 7 4 9
1 6 4 5 9 2 3 8 7
7 9 3 8 1 4 6 2 5
8 2 5 6 7 3 4 9 0

row = 8, col = 8, index = 7, arr[7] = 1
Array: 9 3 4 7 8 2 5 1 6
5 8 6 3 2 1 9 7 4
4 7 2 9 5 8 1 3 6
9 3 1 7 4 6 8 5 2
3 5 7 4 6 9 2 1 8
2 4 9 1 8 7 5 6 3
6 1 8 2 3 5 7 4 9
1 6 4 5 9 2 3 8 7
7 9 3 8 1 4 6 2 5
8 2 5 6 7 3 4 9 1

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