Homework 4

CS201 - Data structures & Algorithms

Spring 2024

This homework is to help you practice the backtracking. We will **not grade your coding style** but you should **make your code as readable as possible**.

Question (Sudoku): You are required to implement a program to solve sudoku puzzles.

Specification: Your task is to implement a class named **"Sudoku_hw4"** that has

1. A public static method **"int[] find_possible_values (int[][] matrix, int x, int y)"** that
    a. gets 3 parameters including (i) matrix is a 9x9 matrix of integers ranging from 0-9 where '0' means an unknown value, (ii) x is an integer from 0-8 representing a row index, (iii) y is an integer from 0-8 representing a column index.
    b. and returns an array of integers ranging from 1 to 9 in ascending order, each element of this array is a possible value of matrix[x][y] (i.e. this value is different from all other non-zero elements in the same row, in the same column, or in the same 3 × 3 submatrices that compose the matrix).
2. A public static method **"void solve (int[][] matrix)"** that gets a 9x9 matrix of integers and fills all zero elements with values ranging from 1-9 to make the matrix become a correct solution (i.e. each column, each row, or each of the nine 3 × 3 submatrices that compose the matrix contains all of the digits from 1 to 9).

Sample test cases:

```
int[][] matrix_1 = {{0, 5, 0, 0, 0, 2, 0, 0, 6},
        {0, 8, 0, 0, 0, 6, 7, 0, 3},
        {0, 0, 0, 7, 4, 0, 0, 0, 8},
        {6, 7, 0, 9, 8, 0, 0, 0, 0},
        {0, 3, 1, 0, 5, 0, 6, 0, 9},
        {0, 0, 0, 0, 0, 3, 8, 2, 0},
        {0, 0, 0, 0, 0, 0, 1, 8, 0},
        {0, 0, 0, 3, 0, 0, 0, 0, 2},
        {9, 2, 0, 5, 0, 0, 0, 7, 0}};
Sudoku_hw4.find_possible_values(matrix_1, 0, 0); // returns [1, 3, 4, 7]
Sudoku_hw4.find_possible_values(matrix_1, 3, 5); // returns [1, 4]
```

```
Sudoku_hw4.find_possible_values(matrix_1, 6, 3); // returns [2, 4, 6]
Sudoku_hw4.solve(matrix_1); // after this call, matrix_1 is the matrix below
// 1 5 7 8 3 2 4 9 6
// 2 8 4 1 9 6 7 5 3
// 3 9 6 7 4 5 2 1 8
// 6 7 2 9 8 4 5 3 1
// 8 3 1 2 5 7 6 4 9
// 5 4 9 6 1 3 8 2 7
// 7 6 3 4 2 9 1 8 5
// 4 1 5 3 7 8 9 6 2
// 9 2 8 5 6 1 3 7 4

//
int[][] matrix_10 = {{5, 3, 0, 0, 8, 0, 0, 0, 0},
        {0, 0, 0, 0, 0, 0, 0, 2, 0},
        {0, 0, 6, 9, 0, 0, 5, 0, 7},
        {0, 0, 7, 5, 0, 0, 4, 0, 9},
        {0, 0, 0, 0, 0, 0, 0, 0, 1},
        {6, 0, 0, 0, 0, 7, 0, 0, 0},
        {0, 0, 4, 0, 0, 0, 0, 1, 0},
        {8, 0, 0, 0, 9, 0, 2, 0, 4},
        {0, 0, 0, 2, 0, 0, 0, 6, 0}};
Sudoku_hw4.solve(matrix_10); // after this call, matrix_10 is the matrix below
// 5 3 2 7 8 4 1 9 6
// 7 4 9 1 5 6 8 2 3
// 1 8 6 9 2 3 5 4 7
// 3 1 7 5 6 2 4 8 9
// 4 2 5 8 3 9 6 7 1
// 6 9 8 4 1 7 3 5 2
// 2 6 4 3 7 8 9 1 5
// 8 7 1 6 9 5 2 3 4
// 9 5 3 2 4 1 7 6 8
```

Implementation & submission:

1. Create a file **"Sudoku_hw4.java"** and add the line

   "package com.gradescope.cs201;"
   at the beginning (otherwise Gradescope can not find your class).
2. Add your code into this file and submit it to Homework 4 on the GradeScope to test your solution.