Lab Session Software Testing 2013, Week 5 With each deliverable, indicate the time spent.

```
module Lab5

where
import Data.List
import Week5
```

Question 1 *The goal of this exercise is to provide the Sudoku program from the lecture notes of this week with a proper specification using Hspec. Rewrite the code using Hspec. Is it also possible to use QuickCheck? If not, why not?*

Question 2 *A Sudoku problem P is minimal if it admits a unique solution, and every problem P' that you can get from P by erasing one of the hints admits more than one solution. How can you test whether the problems generated by the code given in the lecture notes are minimal?*

Deliverables: testing code, test report, indication of time spent.

Question 3 *Write a program that generates Sudoku problems with three empty blocks. Is it also possible to generate Sudoku problems with four empty blocks? Five? How can you check this?*

Question 4 *The goal of this exercise is to extend the Sudoku program from the course notes with functions that can also handle sudokus of a special kind: the sudokus that appear in NRC-Handelsblad each week (designed by Peter Ritmeester, from Oct 8, 2005 onward). These NRC sudokus are special in that they have to satisfy a few extra constraints: in addition to the usual sudoku constraints, each of the 3×3 subgrids with left-top corner (2,2), (2,6), (6,2), and (6,6) should also yield a surjective function.*

Here is an example (the Sudoku exercise of Saturday Nov 26, 2005):

```
+---------+---------+---------+
|         | 3       |         |
|   +-----|--+   +--|-----+   |
|   |     | 7|   |  | 3   |   |
| 2 |     |  |   |  |     | 8 |
+---------+---------+---------+
|   |   6 |  |   |5 |     |   |
|   +-----|--+   +--|-----+   |
|    9  1 | 6       |         |
|   +-----|--+   +--|-----+   |
| 3 |     |  | 7 |1 | 2   |   |
+---------+---------+---------+
|   |     |  |   |  |    3| 1 |
|   |8    |  | 4 |  |     |   |
|   +-----|--+   +--|-----+   |
|       2 |         |         |
+---------+---------+---------+
```

Your task is to formalize this extra constraint, and to use your formalization in a program that can solve this Sudoku.

Deliverables: formal statement of new constraint, modified Haskell program, sudoku solution for the above NRC-Handelsblad sudoku, indication of time spent.

Question 5 *The course notes of this week contain both a Sudoku solver and a Sudoku generator. Use your program from the previous exercise to create a program that generates NRC-Handelsblad Sudoku problems.*

Deliverables: NRC-Handelsblad sudoku generator, indication of time spent.

Question 6 *Can you find a way of classifying the difficulty of a Sudoku problem? Can you modify the Sudoku problem generator so that it can generate problems that are minimal, but easy to solve by hand? Problems that are minimal but hard to solve by hand? How can you test whether the problems your program generates satisfy these properties? Consult [Pelánek(2014)].*

References

[Pelánek(2014)] R. Pelánek. Difficulty rating of sudoku puzzles: An overview and evaluation, 2014.