Course Project No. 5
Обектно Ориентирано Програмиране
(OOP with Java)
Play Sudoku

Acceptable Programming Languages:	**Java 1.8.x**
Deadline:	***(on the final exam date)***
Instructor	**Dr. Evgeny Krustev**

Problem Statement:

Write a RMI **JFrame** client application that **plays the popular Japanese game SUDOKU.** Sudoku puzzles are **9 x 9** grids, and **each square in the grid** consists of a **3 x 3 subgrid** called a **minigrid**. Your goal is to fill in the squares so that each column, row, and minigrid contains the **numbers 1 through 9 exactly once.**

Implement the **Sudoku** game using ***JavaFX*** components (***`GridPane, Menu`*** etc) with the following features:

a) The client **(JavaFX** application)
   - On connecting the ***RMI server*** each client loads a Sudoku puzzle from the server and the game starts
   - **Create and reusable user defined visual component of JavaFX packaged in a JAR file**
   - Ensures that only valid numbers are allowed to be placed in the currently selected cell
   - Checks whether a Sudoku puzzle has been solved
   - Keeps track of the time needed to solve a Sudoku puzzle
   - Undo and redo previous moves
   - Can request and start another game from the server

b) The RMI server (**JavaFX** application)
   - Generates Sudoku puzzles games with 3 levels of difficulty (consult the text sources for this project), where each puzzle should be different than the rest
   - Serves each client by generating and passing to the client a Sudoku puzzle of the selected level of difficulty
   - Presents a solution to the given puzzle at the end of the game, if it is terminated without success by the players.
   - Records statistics in a file about the client username, the level of difficulty and the outcome of the game (total time played and game solved/ unsolved result).

Evaluation:

Your project will be evaluated on the following **general points**:

- **Sophistication/complexity/originality** of the problem being solved/investigated and of the **solution(s)/approaches** considered.
- **Demonstrated ability to extract/analyze** concurrency-related problems/issues from a general problem/area of interest.
- **Clarity of explanations, and for implementations programming skill/quality**. Your report **(in Bulgarian!)** should be well written and free of grammatical and spelling errors. **Programs** must be **well-commented and in a professional style**.
- **Awareness of related work.** Others have considered the same or similar problems before you. Your work does not have to be novel, but you should be able to contextualize your approach. Be sure to explain how each referenced work is *related* to your work. Note that a 5-minute *Google* search will not be adequate; if you are unfamiliar with the required textbooks for the course:
- **Completeness** of the project.

Deliverables: *The files with* :

1. the source code
2. the executable code
3. the instructions for compiling your source code
4. the report explaining the data structures and the algorithm implementation, describe things such as how your code has been tested, limitations of your code, problems encountered, and problems remaining
5. any files used to test the implementation of the program with an explanation about it included in the report.

References:

[1] **P. Deitel, H. Deitel, "**Java How to Program (early objects)**",** Prentice Hall 9 ed. **2012**, ISBN-10: 0132575663, ISBN-13: 978-0-13-257566-9 (**основна)**

[2] **Y. Daniel Liang, "**Introduction to Java Programming**", 7th** ed.,Prentice Hall **2009** ISBN-13: 978-0-13-601267-2

[3] **Bruce Eckel** "Thinking in Java", 4th ed., Prentice Hall 2006 или българското й издание "Да мислим на Java" том 1 и 2, SoftPress, **2001**