**COMP1206 MathDoku Instructions**

**Guide:** This document will help us run and use your application during marking. Please complete the sections below. You may want to include screenshots if this helps explain the functionality. For most sections, 1-2 sentences are probably sufficient.

If you did not implement a particular part, please write “not implemented” in the relevant section.

These instructions are not assessed directly, but they will help ensure that we do not miss any important features of your application.

|  |
| --- |
| **Installing and Running the Application (Part 1)**  *Copy and paste the contents of your README.txt file below.* |
| Mathodoku Coursework  For the coursework I have used the following development environments and libraries:  Java Development Kit 11, Java Run Time Environment 11, javafx library  I have worked on InteliJ Development Environment to build the GUI and its functionality.  The code for the program is divided in three packages: mathdoku.controller, mathdoku.model  and mathdoku.view.  In order to run the program from the console you need to set the path for the javafx library  on your pc and then compile and run the mathdoku.view.View java file  On my PC it looks like this:  "C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2019.2.4\jbr\bin\java.exe"  --module-path "C:\Program Files\Java\javafx-sdk-13.0.2\lib" --add-modules=javafx.controls,javafx.fxml  "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2019.2.4\lib\idea\_rt.jar=65322:C:\Program Files  \JetBrains\IntelliJ IDEA Community Edition 2019.2.4\bin" -Dfile.encoding=UTF-8 -classpath "C:\Users\Home\Desktop  \COMP1206 Programming 2\Coursework\out\production\Coursework;C:\Program Files\Java\javafx-sdk-13.0.2\lib\src.zip;C:  \Program Files\Java\javafx-sdk-13.0.2\lib\javafx-swt.jar;C:\Program Files\Java\javafx-sdk-13.0.2\lib\javafx.web.jar;C:  \Program Files\Java\javafx-sdk-13.0.2\lib\javafx.base.jar;C:\Program Files\Java\javafx-sdk-13.0.2\lib\javafx.fxml.jar;C:  \Program Files\Java\javafx-sdk-13.0.2\lib\javafx.media.jar;C:\Program Files\Java\javafx-sdk-13.0.2\lib\javafx.swing.jar;C:  \Program Files\Java\javafx-sdk-13.0.2\lib\javafx.controls.jar;C:\Program Files\Java\javafx-sdk-13.0.2\lib\javafx.graphics.jar"  mathdoku.view.View |
| **Starting a Game (Optional – Part 1)**  *If any additional steps are needed to start a game, briefly describe them here.* |
|  |
| **Cell Completion (Part 3)**  *Describe how to enter and clear cell values by keyboard and by mouse.* |
| By Keyboard:  After you have selected the cell with your mouse, you enter a digit from 1 to the size of the table by pressing a number from the keyboard. If you want to delete the digit, you press BACKSPACE and cell will go blank as before.  By Mouse:  After you have selected the cell with your mouse, you enter a digit by pressing a button from the button list under the table. In order to delete the value, you need to press the Backspace button right, which is right next to the digit buttons. |
| **Can your application handle - and ÷ cages with more than two cells? (Part 4)** |
| Yes |
| **Mistake Detection (Part 4)**  *Describe how to enable mistake detection in your application.* |
| By clicking the “Operations” menu, you have a list of menu bars: “Show Mistakes” and “Show Hint”. For Mistake Detection click on the Show Mistakes menu bar. |
| **Win Detection / Animation (Parts 4 & 8)**  *Describe how the application notifies the player when the game is won (including any animations you have implemented for Part 8).* |
| After the player has finished his game, the application starts spinning the numbers in all the cells. After the rotations are finished, a fade transition to a new scene is launched. The new scene displays a colourful text: “You Win”. |
| **Clearing (Part 5)**  *Describe how to clear the board.* |
| In order to clear the board, you need to click on the “Edit” menu and then on the “Clear the board” menu bar. |
| **Undo/Redo (Part 5)**  *Describe how to undo / redo actions.* |
| In order to clear the board, you need to click on the “Edit” menu and then on “Undo” or “Redo” menu bars. |
| **Loading Files (Part 6)**  *Describe how to load puzzles both from file and through text input. Also mention any limitations in what puzzles you can load (if any), e.g., up to a certain size if smaller than 8x8.* |
| From File:  Click “New Game” menu and then select “Load from PC” menu bar. A new window will appear. From it you can choose the file you want to be compiled.  From Text:  Click “New Game” menu and then select “Load from text input” menu bar. A new window with a text area will appear. There you can type the input yourself.  Limitations (optional):  No limitations from what I’ve tested so far :) |
| **Font Sizes (Part 7)**  *Describe how to change font sizes* |
| In order to change font size click the “Font” menu and then choose from the following menu bars: “Small”, “Medium” and “Large”. |
| **Solver (Part 9)**  *Describe how to solve a puzzle, how to get a hint and any limitations there might be (e.g., up to what size you can solve reliably and within <1 min). Also mention where we can find your code for solving the puzzle (which files and lines)?* |
| Solve puzzle:  The puzzle is solved automatically after the table is loaded. The only way to see the solved puzzle is to click the hint menu bar until all the cells’ values are shown.  Get hint:  Click the “Operations” menu and then the “Show hint” menu bar. The first cell with wrong or no value will be changed to the right one.  Limitations (optional):  There could be some limitations in some examples with 8x8 tables, in which the program runs > 1min.  Files / lines for solver:  The file Solver.java is the responsible for solving my tables. |
| **Random Game Generator (Part 10)**  *Describe how to generate a random game, including what options the player can select. Also specify where we can find your code for generating the puzzle (which files and lines)? Where in the code do you ensure there is only one solution (which file and lines)?* |
| Generate puzzle (including options):  Click the “New Game” menu and then the “Generate a game” menu bar. An option for the table size will appear: from 2x2 to 8x8.  Files / lines for generator:  In file Controller.java there is a nested class named GenerateGame from line 539 to 902.  File / lines to ensure there is only one solution:  No such |
| **Additional Information (Optional)**  Any other information that may be useful for us to know. |
|  |