**Written self-assessment: MATLAB Programming Project**

**Conceptual Coverage: 40 Points**

For this project, I chose to construct Sudoku in MATLAB. Functionalities of the project are listed on execution.

I believe that I have demonstrated the correct usage of all MATLAB programming concepts mentioned in practicals within the project. My Sudoku game implements the required concepts: for and while loops, (table 1), vectors or 2D arrays (table 2), and conditional execution (table 3).

Table 1:

|  |
| --- |
|  |

Table 2:

|  |
| --- |
|  |

Table 3:

|  |
| --- |
|  |

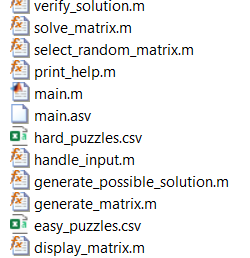
Additionally, the project implements concepts mentioned in practicals such as: functions, Boolean operators, RNG, switch statements, strsplit(), cell arrays of strings / dictionaries, nested loops, figures, mod(), readcsv(), fileread(), lower(), abs(), imshow(), imwrite(), length/size(), flags

It implements other concepts too, such as: global variables, try statements, scalar operations on matrices, find().

(Refer to images below)

|  |
| --- |
| Example of RNG:    Example of Switch conditionals:    Example of strsplit:    Example of cell arrays:    Examples of nested loops:      Example of Figures:    mod(), readcsv(), fileread(), lower(), abs(), imshow(), imwrite(), length(), flags, whos and more, are all present in the implementation too. They can be searched for with a ctrl+f within the project files. |

**Value-Add: 20 Points**

I believe that my project contains many functions of-value that extend beyond the basic functionality of a Sudoku implementation. The solve\_matrix function is one of such functions, as it can solve almost all Sudokus that it was tested with. The generate\_matrix function is another function that I believe is of value, as if the user doesn’t want to play a predefined puzzle, they can generate a random puzzle instead. Additionally, there are functions such as clue() and verify() that can give the user a clue about the value of a cell within the current puzzle or inform the user if they have made a mistake. The user can also import and export their own Sudokus with the save() and load() function.

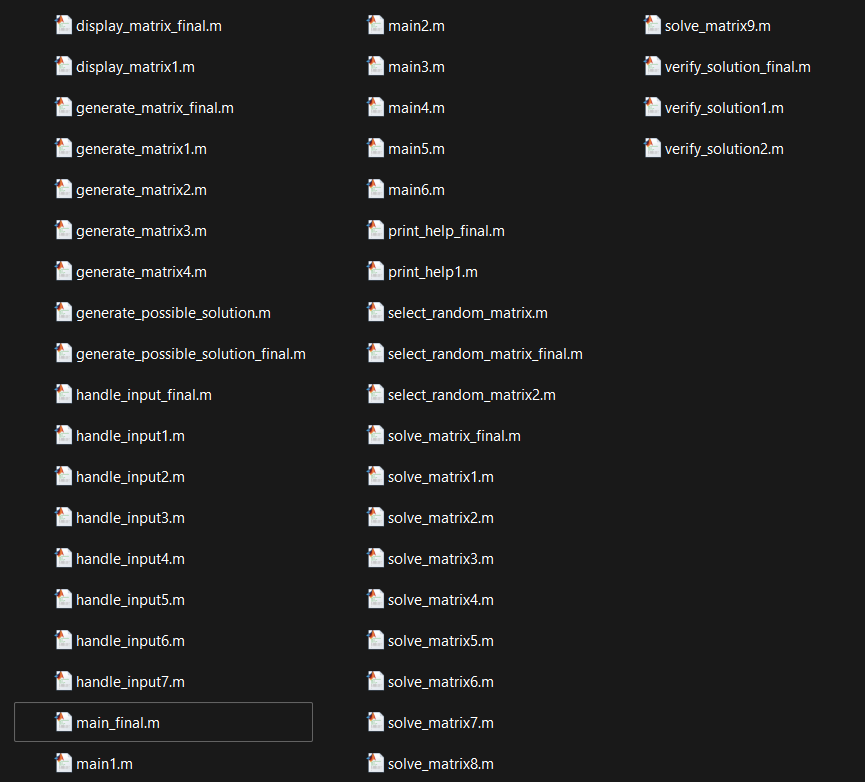
The total line count is 920, 408 of which are comments.

Functions (extending beyond basic functionality) and main file for the project.

The UI:

At the beginning of execution, the user is prompted with a list of commands that they can enter to interact with the program. The arguments entered by the user will be checked for validity, and if the arguments are valid, the appropriate functionality will be executed. The program also clears the command window when certain commands are entered to avoid clutter. The project sanitises user input and has been tested to ensure that it is very hard to break or unbreakable by the user.

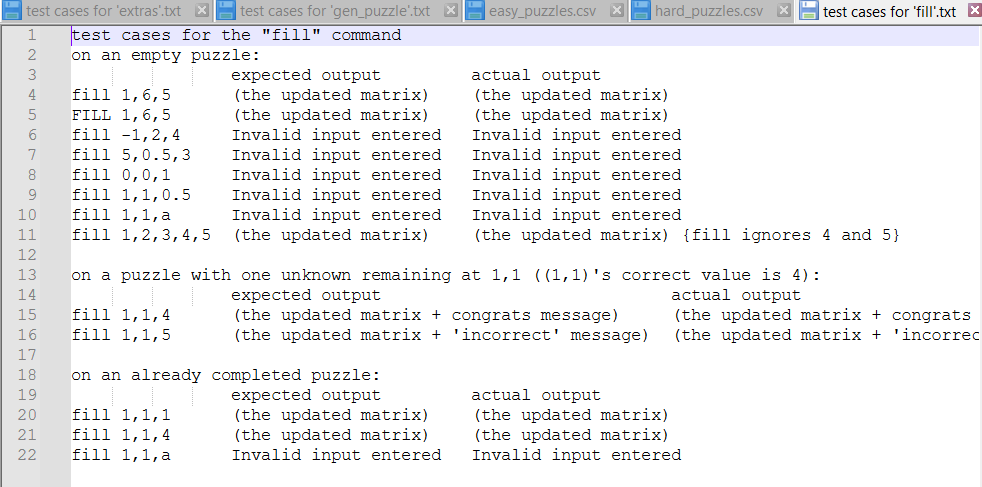
Additionally, I believe that I have demonstrated more creativity through the display of the Sudoku puzzle in the command window by writing a function to space the cells nicely and to print lines around each ‘block’ of the Sudoku puzzle.

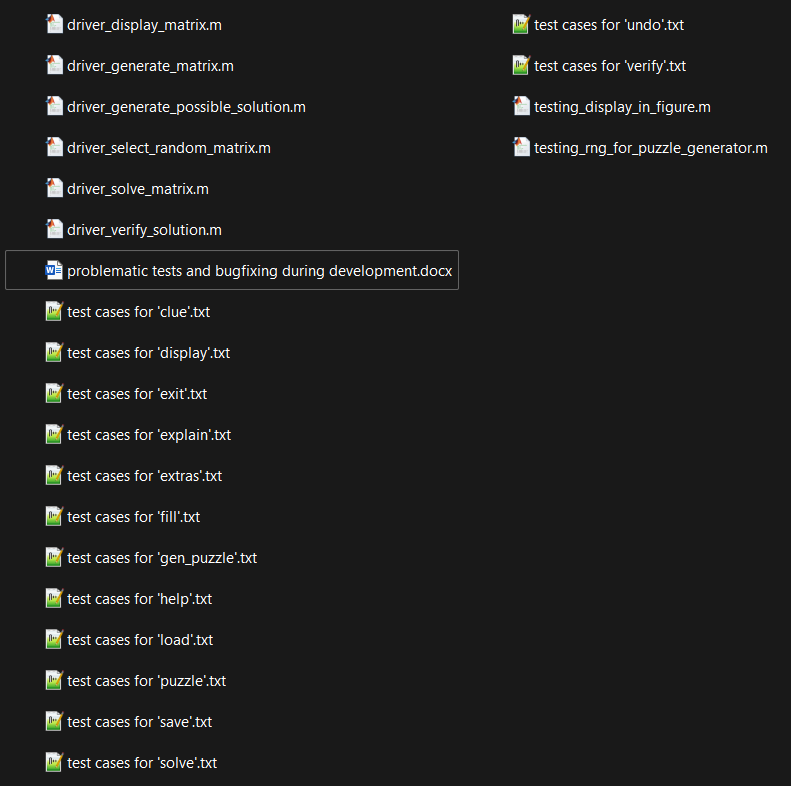
**Incremental Development: 15 Points**

At the beginning of development, I defined each function that I intend to use in the final solution. I then programmed these functions one by one and saved intermediate files with comments that demonstrate the functions’ small increments of development. These intermediate files are all stored in the Incremental Development folder. A screenshot of this folder is shown to the right.

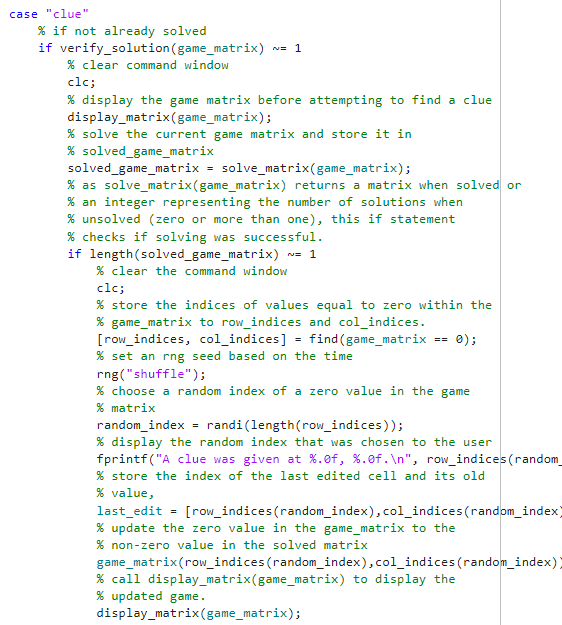
**Testing Strategy 11.25 Points**

The testing folder included in the submission contains test cases for every valid command. These test cases include boundary cases. The testing folder also includes driver files that tests every function that I made with input parameters. The folder also contains tests during incremental development that resulted in finding bugs, however, this document does not contain tests for each function during EVERY stage of development.

A screenshot of the testing folder can be seen below.



**Comments & Style: 7.5 Points**

I believe that I have followed most of the style guidelines and I have consistently used snake case to define variables and functions. I have 408 lines of just comments, however, some of my lines exceed 80 characters and so allocating 10 marks may not be appropriate here.

Images illustrating the comments, snake case, and indents.

