Sodoku Solver:

* The function first imports the sudoku puzzle as a .csv and then converts it into a 2-D Array
* Next box functions are defined
* Then the digit check functions are defined
* First the program scans every cell, and for every cell that is missing a value, a list of the possible values 1-9 is defined.
* Any number that occurs in the row, column, or box of the current cell is removed from the list of possibilities
* If just one possibility remains, the array is updated with the cell set to that value
* This is repeated, either until the puzzle is finished or it has occurred 20 times
* For the next stage, the program does the same, but if there are only two possibilities a cell can take, a 3-d array is set to those possibilities, with the depth dimension being of size 2 to accommodate them
* If a possibility occurs twice in the row, column, or box in cells that have only two possibilities, then it must occur one of them and it can be removed as a possibility in all other cells in the same row column or box
* The program runs the previous algorithm, then does the two digit check to see if this removes any further possibilities
* If a new cell is filled in as a result, the 3D matrix entry for that cell is cleared
* This is repeated until the puzzle is finished or it has run 20 times
* The puzzle then runs the same algorithm, but with 1, then 2, then 3 digits. It was observed that running the 3-digit algorithm in the beginning does not solve the puzzle when a 1 or 2 digit would, and so it was determined to only try it if the lower digit algorithms failed.
* It does the same for 4 digits

Puzzle 1 is easy

Puzzle 2 is one of the hardest puzzles ever made

Puzzle 3 might not be correct

Puzzle 4 is an intermediate

Puzzle 5 is hard

Puzzle 6 is hard, it can’t quite be solved by 3 digits