**Assignment 2: Sudoku**

Student: Mirza Sijamić

Student ID: msc25001

1. Give the representation of a solution (answer) of the problem, as explained during the course. **(1.67)**

I represented the Sudoku solution as an n×n matrix, where n=9 in the standard puzzle, i represents the i-th row and j represents the j-th coulumn of the sudoku board. Each element can take any non-zero positive one digit value, therefore the domain is .

1. Give the equation for the restriction(s) of the problem. **(0.83)**

Let denote the number in the matrix at row I and column j. The constrains can be written as:

For each row i and for any two distinct columns j and k if

For each column j and for any two distinct rows i and k if

For each 3×3 subgrid must not contain matching values from the given domain, if cells (i,j) and (k,l) belong to the same block, then

1. What is considered a state? In addition, explain why. **(0.83)**

State is any board filled with numbers between 1-9 and/or blanks, which are represented as 0 in algorithm. In my representation our state is 9x9 matrix. Each time we change one element of matrix we are getting new state and proceeding with backtracking algorithm to get the solution, getting us closer to the solution without violating any previously mentioned constraints.

1. Which is the initial state? In addition, explain why. **(0.42)**

The starting point is a Sudoku grid with some cells already filled in and others left empty, represented as 0 in given file. The objective is to complete the grid by filling in all the empty cells, ensuring that the final board obeys to all the standard Sudoku rules. The completed board should have no empty spaces and satisfy all the necessary constraints.

1. Which is/are the possible action(s)? In addition, explain why. **(0.42)**

The only possible action is to put a number from the domain into designated place on the board. By doing this we’re changing the state of the board while taking already defined rules into account. If by placing a number we’re violating any of the constrains, the algorithm will backtrack and undo the last move and try another number.

1. What is the maximum branching factor of the tree (b)? In addition, explain why. **(0.42)**

Maximum branching factor of the tree is the worst case scenario and it will be 9. It would occur if we had a cell for example on the sudoku board containing zero constraints such that the entire row i, column j, and the 3 × 3 submatrix are completely empty, In such case we could place all numbers from the domain in .

1. What is the maximum depth of the search tree (m)? In addition, explain why. **(0.41)**

The maximum depth factor would be 81 as the matrix is 9x9. In order for this worst-case scenario to happen, the sudoku board would have to be completely empty, but this never happens. Because of this, the maximum depth factor can vary based on the number of empty cells remaining on the matrix in the given input.