CS480/580 Introduction to Artificial Intelligence

## Assignment 2

Total Points: 100

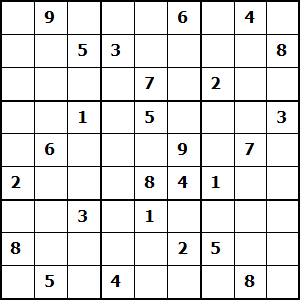
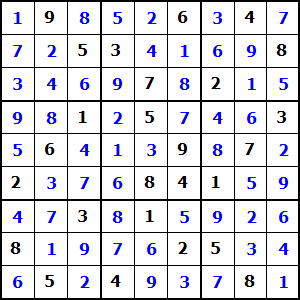
Due Date: 11:59pm on 03/04/2025

**Sudoku as a constraint Satisfaction Problem**

Sudoku is one of the most popular puzzle games of all time. The goal of Sudoku is to fill in a 9×9 grid with digits so that each column, row, and 3×3 section contain the numbers between 1 to 9. At the beginning of the game, the 9×9 grid will have some of the squares filled in. Your job is to use logic to fill in the missing digits and complete the grid. A move is incorrect if:

* Any row contains more than one of the same number from 1 to 9
* Any column contains more than one of the same number from 1 to 9
* Any 3×3 grid contains more than one of the same number from 1 to 9

**An example** is shown below. The grid on the left is a starting configuration, and the grid on the right is the solution. You may not change any of the digits that are in the start configuration. More examples can be found on the following website: <http://www.websudoku.com>.

1. Starting configuration (b) Solved puzzle

For this assignment, you have **4 test puzzles** (easy, medium, difficult and evil puzzles). Note that the label (easy, medium, difficult, and evil) is meant for humans and therefore may not reflect the level of difficulty for a computer program. The following format of the test puzzles can be directly used in Python. You can use any format to represent them in your chosen programming language.

### Easy Puzzle

easy = [[0, 3, 0, 0, 8, 0, 0, 0, 6],

[5, 0, 0, 2, 9, 4, 7, 1, 0],

[0, 0, 0, 3, 0, 0, 5, 0, 0],

[0, 0, 5, 0, 1, 0, 8, 0, 4],

[4, 2, 0, 8, 0, 5, 0, 3, 9],

[1, 0, 8, 0, 3, 0, 6, 0, 0],

[0, 0, 3, 0, 0, 7, 0, 0, 0],

[0, 4, 1, 6, 5, 3, 0, 0, 2],

[2, 0, 0, 0, 4, 0, 0, 6, 0]]

### Medium puzzle

medium = [[3, 0, 8, 2, 9, 6, 0, 0, 0],

[0, 4, 0, 0, 0, 8, 0, 0, 0],

[5, 0, 2, 1, 0, 0, 0, 8, 7],

[0, 1, 3, 0, 0, 0, 0, 0, 0],

[7, 8, 0, 0, 0, 0, 0, 3, 5],

[0, 0, 0, 0, 0, 0, 4, 1, 0],

[1, 2, 0, 0, 0, 7, 8, 0, 3],

[0, 0, 0, 8, 0, 0, 0, 2, 0],

[0, 0, 0, 5, 4, 2, 1, 0, 6]]

### Hard Puzzle

hard = [[7, 0, 0, 0, 0, 0, 0, 0, 0],

[6, 0, 0, 4, 1, 0, 2, 5, 0],

[0, 1, 3, 0, 9, 5, 0, 0, 0],

[8, 6, 0, 0, 0, 0, 0, 0, 0],

[3, 0, 1, 0, 0, 0, 4, 0, 5],

[0, 0, 0, 0, 0, 0, 0, 8, 6],

[0, 0, 0, 8, 4, 0, 5, 3, 0],

[0, 4, 2, 0, 3, 6, 0, 0, 7],

[0, 0, 0, 0, 0, 0, 0, 0, 9]]

### EVIL Puzzle

evil = [[0, 6, 0, 8, 0, 0, 0, 0, 0],

[0, 0, 4, 0, 6, 0, 0, 0, 9],

[1, 0, 0, 0, 4, 3, 0, 6, 0],

[0, 5, 2, 0, 0, 0, 0, 0, 0],

[0, 0, 8, 6, 0, 9, 3, 0, 0],

[0, 0, 0, 0, 0, 0, 5, 7, 0],

[0, 1, 0, 4, 8, 0, 0, 0, 5],

[8, 0, 0, 0, 1, 0, 2, 0, 0],

[0, 0, 0, 0, 0, 5, 0, 4, 0]]

For this programming assignment, your tasks are to write an AI program to solve the 4 Sudoku test puzzles as constraint satisfaction problems.

**Task1** (10 points): Formulate the Sudoku puzzle as a constraint satisfaction problem (CSP)

A detailed description of all variables, domains and constraints that are sufficient to model Sudoku as a CSP.

**Task2** (40 points): Implement a simple backtracking algorithm to solve Sudoku based on the formulation.

The selection of variables and assignment of values can be done either in order or randomly.

**Task3** (40 points): Implement a smart backtracking algorithm to solve Sudoku based on the formulation.

Incorporate at least one strategy of minimum remaining values (MRV), degree heuristic, least constraining value (LCV), and forward checking (FC) in your backtracking algorithm.

**Task4** (10 points): Report and Analysis

Analyze the performance of your Sudoku solvers (one basic backtracking algorithm and one smart backtracking algorithm) on these puzzles, including the running time and the number of trials of assigning values to the variables.

**What to Hand in**

1. Formulation of the Sudoku CSP problems.
2. Well documented codes implementing the Sudoku solvers. A README file should provide instructions on how to compile and execute the code.
3. For both Task2 and Task3, please provide the screenshots of the solutions, the running time, and the number of the total value assignments to the variables.
4. Analysis of your program. Please compare the time and the number of trials of assigning values to the variables.

Please submit a zip file with the program and the analysis report on Canvas before the assignment due date.

**Programming Language**

You can choose any programming language.