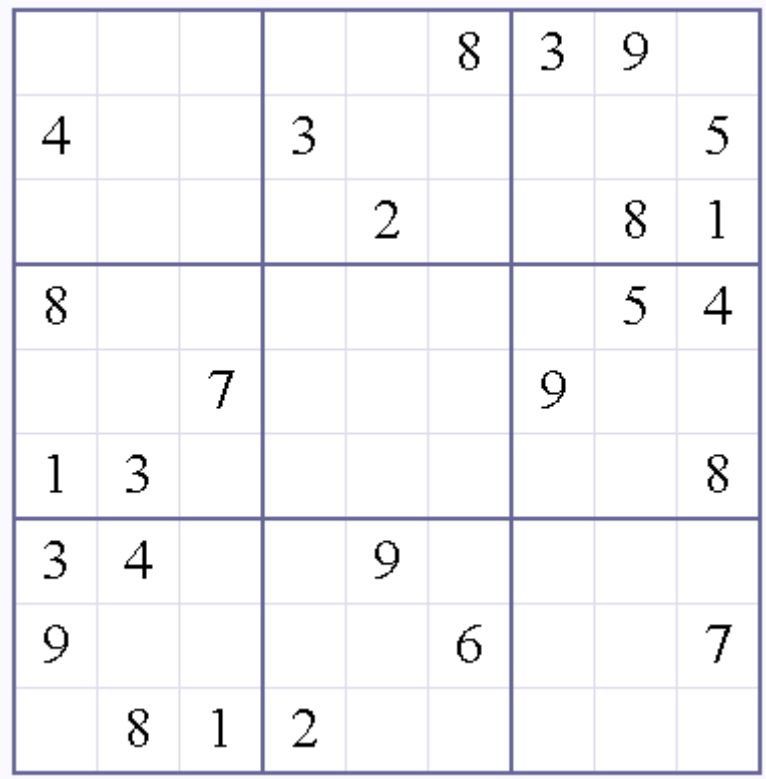
**CS480/580 Introduction to Artificial Intelligence (Fall, 2017)**
**Assignment 3**

Due: Thursday, Oct. 26, 2017
Total Points: 100

1. 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



| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | 8 | 3 | 9 | |
| 4 | | | 3 | | | | | 5 |
| | | | | 2 | | | 8 | 1 |
| 8 | | | | | | | 5 | 4 |
| | | 7 | | | | 9 | | |
| 1 | 3 | | | | | | | 8 |
| 3 | 4 | | | 9 | | | | |
| 9 | | | | | 6 | | | 7 |
| | 8 | 1 | 2 | | | | | |

A Sudoku text file should contain the information of the Sudoku puzzle in the following format.

```
Sudoku 01
003020600
900305001
001806400
008102900
700000008
006708200
002609500
800203009
005010300
```

Your task is to write an AI program to solve the Sudoku puzzle as a constraint satisfaction problem.

Task 1. (Parse the Sudoku data file) (5 pts)

Task 2. (Naïve Backtracking Algorithm) (30 pts)

Implement a naïve backtracking algorithm. The selection of variables and assignment of values can be done either in order or randomly.

Task 3. (Smart Backtracking Algorithm) (40 pts)

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

Task 4. (Report and Analysis) (25 pts)

The following website provides Sudoku puzzles in levels of easy, medium, hard, and evil. Analyze the performance of your Sudoku solver on these puzzles.

http://www.websodoku.com

What to Hand in

1. Well documented codes implementing your Sudoku solver. A README file should provide instructions on how to compile and execute the code.
2. Solutions you obtained, describing your methods, and analysis in tasks.

Please turn in your written part in class and send the programs to dfeng@cs.odu.edu before the assignment due date.