# CSE 537 Home Assignment 3

Project Report

**Backtracking Approach**

This is a naïve brute force depth first search approach which will try out all possibilities.

**Backtracking + MRV**

This method uses an extra heuristic such that the cells with least number of possible values are filled first. By this heuristic we could eliminate many branches that would fill up these constraints last and probably lead to failure.

**Backtracking + MRV + Forward Checking**

This method uses a check to see whether a particular assignment which would definitely lead to a failure. By this method we prune search branches which are going to fail eventually

**Backtracking + MRV + CP**

Forward checking just checks for failures in a single search branch, but Constraint propagation checks for failures in all the search branches. The number of branches pruned by this method is generally higher than that of forward checking

**Min-Conflicts**

This is an iterative algorithm unlike other recursive algorithms. This algorithm picks a random empty cell and calculates the value of variable which leads to minimum conflicts. The algorithm is bounded by a limited number of iterations.

**Preferred Algorithm**

When we have to give out a solution in limited time the **Min-Conflicts** algorithm is the best fit as it has a fixed worst case running time.

Among the recursive algorithms the **Backtracking + MRV + CP** algorithm is the best fit as it prunes maximum search space. It checks for consistency of assignment before every assignment which leads to effective pruning of search trees.

==========================================================================

Some of outputs on which the game was tested are given below:

**python sudoku.py --input game-minconflict.txt**

Backtracking:

Execution Time: 0.000429

Consistency Checks: 29

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['2 ', '4 ', '3 ', '1 ']

['3 ', '1 ', '2 ', '4 ']

['1 ', '3 ', '4 ', '2 ']

['4 ', '2 ', '1 ', '3 ']

backtrackingMRV:

Execution Time: 0.002887

Consistency Checks: 245

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['2 ', '4 ', '3 ', '1 ']

['3 ', '1 ', '2 ', '4 ']

['1 ', '3 ', '4 ', '2 ']

['4 ', '2 ', '1 ', '3 ']

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['2 ', '4 ', '3 ', '1 ']

['3 ', '1 ', '2 ', '4 ']

['1 ', '3 ', '4 ', '2 ']

['4 ', '2 ', '1 ', '3 ']

backtrackingMRVfwd:

Execution Time: 0.009674

Consistency Checks: 260

Solution printed above:

backtrackingMRVcp:

Execution Time: 0.021833

Consistency Checks: 245

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['2 ', '4 ', '3 ', '1 ']

['3 ', '1 ', '2 ', '4 ']

['1 ', '3 ', '4 ', '2 ']

['4 ', '2 ', '1 ', '3 ']

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['2 ', '4 ', '3 ', '1 ']

['3 ', '1 ', '2 ', '4 ']

['1 ', '3 ', '4 ', '2 ']

['4 ', '2 ', '1 ', '3 ']

minConflict:

Execution Time: 0.110744

Consistency Checks: 10

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['2 ', '4 ', '3 ', '1 ']

['3 ', '1 ', '2 ', '4 ']

['1 ', '3 ', '4 ', '2 ']

['4 ', '2 ', '1 ', '3 ']

==========================================================================

**python sudoku.py --input game1.txt**

Backtracking:

Execution Time: 0.03378

Consistency Checks: 2074

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['1 ', '9 ', '6 ', '7 ', '11 ', '8 ', '5 ', '4 ', '2 ', '10 ', '12 ', '3 ']

['5 ', '2 ', '10 ', '11 ', '6 ', '12 ', '1 ', '3 ', '7 ', '8 ', '4 ', '9 ']

['12 ', '8 ', '3 ', '4 ', '2 ', '7 ', '10 ', '9 ', '11 ', '1 ', '6 ', '5 ']

['2 ', '7 ', '9 ', '1 ', '4 ', '10 ', '8 ', '5 ', '12 ', '6 ', '3 ', '11 ']

['11 ', '12 ', '8 ', '10 ', '3 ', '1 ', '6 ', '2 ', '5 ', '9 ', '7 ', '4 ']

['3 ', '5 ', '4 ', '6 ', '9 ', '11 ', '7 ', '12 ', '8 ', '2 ', '10 ', '1 ']

['9 ', '10 ', '2 ', '12 ', '8 ', '3 ', '4 ', '11 ', '6 ', '5 ', '1 ', '7 ']

['6 ', '3 ', '1 ', '8 ', '12 ', '5 ', '9 ', '7 ', '10 ', '4 ', '11 ', '2 ']

['4 ', '11 ', '7 ', '5 ', '1 ', '6 ', '2 ', '10 ', '3 ', '12 ', '9 ', '8 ']

['10 ', '1 ', '12 ', '2 ', '7 ', '4 ', '11 ', '8 ', '9 ', '3 ', '5 ', '6 ']

['8 ', '4 ', '11 ', '3 ', '5 ', '9 ', '12 ', '6 ', '1 ', '7 ', '2 ', '10 ']

['7 ', '6 ', '5 ', '9 ', '10 ', '2 ', '3 ', '1 ', '4 ', '11 ', '8 ', '12 ']

backtrackingMRV:

Execution Time: 0.587852

Consistency Checks: 35590

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['1 ', '9 ', '6 ', '7 ', '11 ', '8 ', '5 ', '4 ', '2 ', '10 ', '12 ', '3 ']

['5 ', '2 ', '10 ', '11 ', '6 ', '12 ', '1 ', '3 ', '7 ', '8 ', '4 ', '9 ']

['12 ', '8 ', '3 ', '4 ', '2 ', '7 ', '10 ', '9 ', '11 ', '1 ', '6 ', '5 ']

['2 ', '7 ', '9 ', '1 ', '4 ', '10 ', '8 ', '5 ', '12 ', '6 ', '3 ', '11 ']

['11 ', '12 ', '8 ', '10 ', '3 ', '1 ', '6 ', '2 ', '5 ', '9 ', '7 ', '4 ']

['3 ', '5 ', '4 ', '6 ', '9 ', '11 ', '7 ', '12 ', '8 ', '2 ', '10 ', '1 ']

['9 ', '10 ', '2 ', '12 ', '8 ', '3 ', '4 ', '11 ', '6 ', '5 ', '1 ', '7 ']

['6 ', '3 ', '1 ', '8 ', '12 ', '5 ', '9 ', '7 ', '10 ', '4 ', '11 ', '2 ']

['4 ', '11 ', '7 ', '5 ', '1 ', '6 ', '2 ', '10 ', '3 ', '12 ', '9 ', '8 ']

['10 ', '1 ', '12 ', '2 ', '7 ', '4 ', '11 ', '8 ', '9 ', '3 ', '5 ', '6 ']

['8 ', '4 ', '11 ', '3 ', '5 ', '9 ', '12 ', '6 ', '1 ', '7 ', '2 ', '10 ']

['7 ', '6 ', '5 ', '9 ', '10 ', '2 ', '3 ', '1 ', '4 ', '11 ', '8 ', '12 ']

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['1 ', '9 ', '6 ', '7 ', '11 ', '8 ', '5 ', '4 ', '2 ', '10 ', '12 ', '3 ']

['5 ', '2 ', '10 ', '11 ', '6 ', '12 ', '1 ', '3 ', '7 ', '8 ', '4 ', '9 ']

['12 ', '8 ', '3 ', '4 ', '2 ', '7 ', '10 ', '9 ', '11 ', '1 ', '6 ', '5 ']

['2 ', '7 ', '9 ', '1 ', '4 ', '10 ', '8 ', '5 ', '12 ', '6 ', '3 ', '11 ']

['11 ', '12 ', '8 ', '10 ', '3 ', '1 ', '6 ', '2 ', '5 ', '9 ', '7 ', '4 ']

['3 ', '5 ', '4 ', '6 ', '9 ', '11 ', '7 ', '12 ', '8 ', '2 ', '10 ', '1 ']

['9 ', '10 ', '2 ', '12 ', '8 ', '3 ', '4 ', '11 ', '6 ', '5 ', '1 ', '7 ']

['6 ', '3 ', '1 ', '8 ', '12 ', '5 ', '9 ', '7 ', '10 ', '4 ', '11 ', '2 ']

['4 ', '11 ', '7 ', '5 ', '1 ', '6 ', '2 ', '10 ', '3 ', '12 ', '9 ', '8 ']

['10 ', '1 ', '12 ', '2 ', '7 ', '4 ', '11 ', '8 ', '9 ', '3 ', '5 ', '6 ']

['8 ', '4 ', '11 ', '3 ', '5 ', '9 ', '12 ', '6 ', '1 ', '7 ', '2 ', '10 ']

['7 ', '6 ', '5 ', '9 ', '10 ', '2 ', '3 ', '1 ', '4 ', '11 ', '8 ', '12 ']

backtrackingMRVfwd:

Execution Time: 0.803733

Consistency Checks: 36024

Solution printed above

backtrackingMRVcp:

Execution Time: 0.506625

Consistency Checks: 35590

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['1 ', '9 ', '6 ', '7 ', '11 ', '8 ', '5 ', '4 ', '2 ', '10 ', '12 ', '3 ']

['5 ', '2 ', '10 ', '11 ', '6 ', '12 ', '1 ', '3 ', '7 ', '8 ', '4 ', '9 ']

['12 ', '8 ', '3 ', '4 ', '2 ', '7 ', '10 ', '9 ', '11 ', '1 ', '6 ', '5 ']

['2 ', '7 ', '9 ', '1 ', '4 ', '10 ', '8 ', '5 ', '12 ', '6 ', '3 ', '11 ']

['11 ', '12 ', '8 ', '10 ', '3 ', '1 ', '6 ', '2 ', '5 ', '9 ', '7 ', '4 ']

['3 ', '5 ', '4 ', '6 ', '9 ', '11 ', '7 ', '12 ', '8 ', '2 ', '10 ', '1 ']

['9 ', '10 ', '2 ', '12 ', '8 ', '3 ', '4 ', '11 ', '6 ', '5 ', '1 ', '7 ']

['6 ', '3 ', '1 ', '8 ', '12 ', '5 ', '9 ', '7 ', '10 ', '4 ', '11 ', '2 ']

['4 ', '11 ', '7 ', '5 ', '1 ', '6 ', '2 ', '10 ', '3 ', '12 ', '9 ', '8 ']

['10 ', '1 ', '12 ', '2 ', '7 ', '4 ', '11 ', '8 ', '9 ', '3 ', '5 ', '6 ']

['8 ', '4 ', '11 ', '3 ', '5 ', '9 ', '12 ', '6 ', '1 ', '7 ', '2 ', '10 ']

['7 ', '6 ', '5 ', '9 ', '10 ', '2 ', '3 ', '1 ', '4 ', '11 ', '8 ', '12 ']

minConflict:

Execution Time: 15.172221

Consistency Checks: 100000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['1 ', '9 ', '7 ', '0 ', '11 ', '8 ', '2 ', '4 ', '0 ', '10 ', '12 ', '3 ']

['5 ', '2 ', '10 ', '11 ', '6 ', '12 ', '1 ', '3 ', '7 ', '8 ', '4 ', '9 ']

['12 ', '8 ', '3 ', '4 ', '0 ', '7 ', '10 ', '9 ', '2 ', '1 ', '6 ', '5 ']

['2 ', '7 ', '9 ', '1 ', '4 ', '10 ', '8 ', '5 ', '12 ', '6 ', '3 ', '11 ']

['11 ', '12 ', '8 ', '6 ', '3 ', '1 ', '0 ', '2 ', '5 ', '9 ', '7 ', '4 ']

['3 ', '5 ', '4 ', '0 ', '9 ', '11 ', '6 ', '12 ', '0 ', '2 ', '10 ', '8 ']

['9 ', '10 ', '2 ', '12 ', '0 ', '0 ', '3 ', '11 ', '1 ', '5 ', '8 ', '7 ']

['6 ', '3 ', '1 ', '8 ', '12 ', '5 ', '4 ', '7 ', '10 ', '11 ', '0 ', '2 ']

['0 ', '11 ', '0 ', '5 ', '1 ', '6 ', '0 ', '10 ', '3 ', '4 ', '9 ', '0 ']

['10 ', '1 ', '6 ', '2 ', '7 ', '3 ', '11 ', '8 ', '9 ', '0 ', '5 ', '0 ']

['7 ', '4 ', '11 ', '3 ', '5 ', '9 ', '12 ', '6 ', '8 ', '0 ', '2 ', '1 ']

['8 ', '0 ', '5 ', '9 ', '10 ', '2 ', '0 ', '1 ', '4 ', '3 ', '11 ', '12 ']

==========================================================================

**python sudoku.py --input game2.txt**

Backtracking:

Execution Time: 6.222218

Consistency Checks: 551734

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['3 ', '2 ', '12 ', '11 ', '7 ', '10 ', '1 ', '4 ', '6 ', '8 ', '9 ', '5 ']

['5 ', '9 ', '8 ', '6 ', '3 ', '2 ', '12 ', '11 ', '7 ', '4 ', '1 ', '10 ']

['10 ', '1 ', '4 ', '7 ', '5 ', '8 ', '6 ', '9 ', '2 ', '3 ', '11 ', '12 ']

['2 ', '12 ', '11 ', '5 ', '1 ', '7 ', '8 ', '3 ', '10 ', '6 ', '4 ', '9 ']

['9 ', '8 ', '6 ', '10 ', '2 ', '4 ', '11 ', '12 ', '5 ', '7 ', '3 ', '1 ']

['4 ', '7 ', '3 ', '1 ', '6 ', '9 ', '5 ', '10 ', '11 ', '12 ', '2 ', '8 ']

['11 ', '5 ', '9 ', '12 ', '8 ', '6 ', '10 ', '2 ', '3 ', '1 ', '7 ', '4 ']

['7 ', '3 ', '1 ', '2 ', '11 ', '12 ', '4 ', '5 ', '9 ', '10 ', '8 ', '6 ']

['8 ', '6 ', '10 ', '4 ', '9 ', '1 ', '3 ', '7 ', '12 ', '11 ', '5 ', '2 ']

['12 ', '4 ', '2 ', '9 ', '10 ', '3 ', '7 ', '8 ', '1 ', '5 ', '6 ', '11 ']

['1 ', '10 ', '5 ', '3 ', '4 ', '11 ', '9 ', '6 ', '8 ', '2 ', '12 ', '7 ']

['6 ', '11 ', '7 ', '8 ', '12 ', '5 ', '2 ', '1 ', '4 ', '9 ', '10 ', '3 ']

backtrackingMRV:

Execution Time: 42.66895

Consistency Checks: 3263734

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['3 ', '2 ', '12 ', '11 ', '7 ', '10 ', '1 ', '4 ', '6 ', '8 ', '9 ', '5 ']

['5 ', '9 ', '8 ', '6 ', '3 ', '2 ', '12 ', '11 ', '7 ', '4 ', '1 ', '10 ']

['10 ', '1 ', '4 ', '7 ', '5 ', '8 ', '6 ', '9 ', '2 ', '3 ', '11 ', '12 ']

['2 ', '12 ', '11 ', '5 ', '1 ', '7 ', '8 ', '3 ', '10 ', '6 ', '4 ', '9 ']

['9 ', '8 ', '6 ', '10 ', '2 ', '4 ', '11 ', '12 ', '5 ', '7 ', '3 ', '1 ']

['4 ', '7 ', '3 ', '1 ', '6 ', '9 ', '5 ', '10 ', '11 ', '12 ', '2 ', '8 ']

['11 ', '5 ', '9 ', '12 ', '8 ', '6 ', '10 ', '2 ', '3 ', '1 ', '7 ', '4 ']

['7 ', '3 ', '1 ', '2 ', '11 ', '12 ', '4 ', '5 ', '9 ', '10 ', '8 ', '6 ']

['8 ', '6 ', '10 ', '4 ', '9 ', '1 ', '3 ', '7 ', '12 ', '11 ', '5 ', '2 ']

['12 ', '4 ', '2 ', '9 ', '10 ', '3 ', '7 ', '8 ', '1 ', '5 ', '6 ', '11 ']

['1 ', '10 ', '5 ', '3 ', '4 ', '11 ', '9 ', '6 ', '8 ', '2 ', '12 ', '7 ']

['6 ', '11 ', '7 ', '8 ', '12 ', '5 ', '2 ', '1 ', '4 ', '9 ', '10 ', '3 ']

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['3 ', '2 ', '12 ', '11 ', '7 ', '10 ', '1 ', '4 ', '6 ', '8 ', '9 ', '5 ']

['5 ', '9 ', '8 ', '6 ', '3 ', '2 ', '12 ', '11 ', '7 ', '4 ', '1 ', '10 ']

['10 ', '1 ', '4 ', '7 ', '5 ', '8 ', '6 ', '9 ', '2 ', '3 ', '11 ', '12 ']

['2 ', '12 ', '11 ', '5 ', '1 ', '7 ', '8 ', '3 ', '10 ', '6 ', '4 ', '9 ']

['9 ', '8 ', '6 ', '10 ', '2 ', '4 ', '11 ', '12 ', '5 ', '7 ', '3 ', '1 ']

['4 ', '7 ', '3 ', '1 ', '6 ', '9 ', '5 ', '10 ', '11 ', '12 ', '2 ', '8 ']

['11 ', '5 ', '9 ', '12 ', '8 ', '6 ', '10 ', '2 ', '3 ', '1 ', '7 ', '4 ']

['7 ', '3 ', '1 ', '2 ', '11 ', '12 ', '4 ', '5 ', '9 ', '10 ', '8 ', '6 ']

['8 ', '6 ', '10 ', '4 ', '9 ', '1 ', '3 ', '7 ', '12 ', '11 ', '5 ', '2 ']

['12 ', '4 ', '2 ', '9 ', '10 ', '3 ', '7 ', '8 ', '1 ', '5 ', '6 ', '11 ']

['1 ', '10 ', '5 ', '3 ', '4 ', '11 ', '9 ', '6 ', '8 ', '2 ', '12 ', '7 ']

['6 ', '11 ', '7 ', '8 ', '12 ', '5 ', '2 ', '1 ', '4 ', '9 ', '10 ', '3 ']

backtrackingMRVfwd:

Execution Time: 12.465929

Consistency Checks: 580308

Solution printed above

backtrackingMRVcp:

Execution Time: 47.237314

Consistency Checks: 3263734

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['3 ', '2 ', '12 ', '11 ', '7 ', '10 ', '1 ', '4 ', '6 ', '8 ', '9 ', '5 ']

['5 ', '9 ', '8 ', '6 ', '3 ', '2 ', '12 ', '11 ', '7 ', '4 ', '1 ', '10 ']

['10 ', '1 ', '4 ', '7 ', '5 ', '8 ', '6 ', '9 ', '2 ', '3 ', '11 ', '12 ']

['2 ', '12 ', '11 ', '5 ', '1 ', '7 ', '8 ', '3 ', '10 ', '6 ', '4 ', '9 ']

['9 ', '8 ', '6 ', '10 ', '2 ', '4 ', '11 ', '12 ', '5 ', '7 ', '3 ', '1 ']

['4 ', '7 ', '3 ', '1 ', '6 ', '9 ', '5 ', '10 ', '11 ', '12 ', '2 ', '8 ']

['11 ', '5 ', '9 ', '12 ', '8 ', '6 ', '10 ', '2 ', '3 ', '1 ', '7 ', '4 ']

['7 ', '3 ', '1 ', '2 ', '11 ', '12 ', '4 ', '5 ', '9 ', '10 ', '8 ', '6 ']

['8 ', '6 ', '10 ', '4 ', '9 ', '1 ', '3 ', '7 ', '12 ', '11 ', '5 ', '2 ']

['12 ', '4 ', '2 ', '9 ', '10 ', '3 ', '7 ', '8 ', '1 ', '5 ', '6 ', '11 ']

['1 ', '10 ', '5 ', '3 ', '4 ', '11 ', '9 ', '6 ', '8 ', '2 ', '12 ', '7 ']

['6 ', '11 ', '7 ', '8 ', '12 ', '5 ', '2 ', '1 ', '4 ', '9 ', '10 ', '3 ']

minConflict:

Execution Time: 16.634252

Consistency Checks: 100000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['0 ', '2 ', '12 ', '11 ', '0 ', '10 ', '1 ', '4 ', '9 ', '8 ', '7 ', '5 ']

['0 ', '9 ', '8 ', '5 ', '3 ', '2 ', '7 ', '0 ', '0 ', '4 ', '1 ', '10 ']

['10 ', '1 ', '4 ', '0 ', '5 ', '8 ', '6 ', '9 ', '0 ', '11 ', '3 ', '12 ']

['2 ', '8 ', '10 ', '0 ', '1 ', '7 ', '5 ', '11 ', '6 ', '3 ', '4 ', '9 ']

['9 ', '0 ', '6 ', '1 ', '2 ', '0 ', '8 ', '12 ', '5 ', '7 ', '0 ', '0 ']

['4 ', '0 ', '3 ', '7 ', '6 ', '9 ', '0 ', '10 ', '11 ', '12 ', '2 ', '1 ']

['3 ', '5 ', '11 ', '12 ', '8 ', '6 ', '10 ', '2 ', '7 ', '1 ', '0 ', '4 ']

['7 ', '0 ', '1 ', '2 ', '11 ', '4 ', '12 ', '5 ', '3 ', '10 ', '9 ', '6 ']

['0 ', '4 ', '9 ', '6 ', '0 ', '1 ', '3 ', '7 ', '8 ', '2 ', '5 ', '0 ']

['12 ', '0 ', '2 ', '9 ', '10 ', '3 ', '4 ', '8 ', '1 ', '5 ', '6 ', '11 ']

['1 ', '10 ', '5 ', '3 ', '7 ', '11 ', '9 ', '6 ', '2 ', '0 ', '12 ', '8 ']

['6 ', '11 ', '7 ', '4 ', '12 ', '5 ', '2 ', '1 ', '0 ', '9 ', '10 ', '3 ']

==========================================================================

**python sudoku.py --input game6.txt**

Backtracking:

Execution Time: 2.963742

Consistency Checks: 287952

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['4 ', '2 ', '7 ', '5 ', '3 ', '9 ', '8 ', '6 ', '1 ']

['9 ', '5 ', '1 ', '4 ', '8 ', '6 ', '3 ', '2 ', '7 ']

['8 ', '6 ', '3 ', '7 ', '1 ', '2 ', '9 ', '5 ', '4 ']

['1 ', '3 ', '8 ', '9 ', '4 ', '5 ', '2 ', '7 ', '6 ']

['2 ', '9 ', '5 ', '1 ', '6 ', '7 ', '4 ', '3 ', '8 ']

['6 ', '7 ', '4 ', '8 ', '2 ', '3 ', '1 ', '9 ', '5 ']

['3 ', '8 ', '6 ', '2 ', '5 ', '1 ', '7 ', '4 ', '9 ']

['5 ', '1 ', '9 ', '3 ', '7 ', '4 ', '6 ', '8 ', '2 ']

['7 ', '4 ', '2 ', '6 ', '9 ', '8 ', '5 ', '1 ', '3 ']

backtrackingMRV:

Execution Time: 0.261263

Consistency Checks: 20193

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['4 ', '2 ', '7 ', '5 ', '3 ', '9 ', '8 ', '6 ', '1 ']

['9 ', '5 ', '1 ', '4 ', '8 ', '6 ', '3 ', '2 ', '7 ']

['8 ', '6 ', '3 ', '7 ', '1 ', '2 ', '9 ', '5 ', '4 ']

['1 ', '3 ', '8 ', '9 ', '4 ', '5 ', '2 ', '7 ', '6 ']

['2 ', '9 ', '5 ', '1 ', '6 ', '7 ', '4 ', '3 ', '8 ']

['6 ', '7 ', '4 ', '8 ', '2 ', '3 ', '1 ', '9 ', '5 ']

['3 ', '8 ', '6 ', '2 ', '5 ', '1 ', '7 ', '4 ', '9 ']

['5 ', '1 ', '9 ', '3 ', '7 ', '4 ', '6 ', '8 ', '2 ']

['7 ', '4 ', '2 ', '6 ', '9 ', '8 ', '5 ', '1 ', '3 ']

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['4 ', '2 ', '7 ', '5 ', '3 ', '9 ', '8 ', '6 ', '1 ']

['9 ', '5 ', '1 ', '4 ', '8 ', '6 ', '3 ', '2 ', '7 ']

['8 ', '6 ', '3 ', '7 ', '1 ', '2 ', '9 ', '5 ', '4 ']

['1 ', '3 ', '8 ', '9 ', '4 ', '5 ', '2 ', '7 ', '6 ']

['2 ', '9 ', '5 ', '1 ', '6 ', '7 ', '4 ', '3 ', '8 ']

['6 ', '7 ', '4 ', '8 ', '2 ', '3 ', '1 ', '9 ', '5 ']

['3 ', '8 ', '6 ', '2 ', '5 ', '1 ', '7 ', '4 ', '9 ']

['5 ', '1 ', '9 ', '3 ', '7 ', '4 ', '6 ', '8 ', '2 ']

['7 ', '4 ', '2 ', '6 ', '9 ', '8 ', '5 ', '1 ', '3 ']

backtrackingMRVfwd:

Execution Time: 4.782954

Consistency Checks: 205848

Solution printed above

backtrackingMRVcp:

Execution Time: 0.372151

Consistency Checks: 20193

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['4 ', '2 ', '7 ', '5 ', '3 ', '9 ', '8 ', '6 ', '1 ']

['9 ', '5 ', '1 ', '4 ', '8 ', '6 ', '3 ', '2 ', '7 ']

['8 ', '6 ', '3 ', '7 ', '1 ', '2 ', '9 ', '5 ', '4 ']

['1 ', '3 ', '8 ', '9 ', '4 ', '5 ', '2 ', '7 ', '6 ']

['2 ', '9 ', '5 ', '1 ', '6 ', '7 ', '4 ', '3 ', '8 ']

['6 ', '7 ', '4 ', '8 ', '2 ', '3 ', '1 ', '9 ', '5 ']

['3 ', '8 ', '6 ', '2 ', '5 ', '1 ', '7 ', '4 ', '9 ']

['5 ', '1 ', '9 ', '3 ', '7 ', '4 ', '6 ', '8 ', '2 ']

['7 ', '4 ', '2 ', '6 ', '9 ', '8 ', '5 ', '1 ', '3 ']

minConflict:

Execution Time: 11.394667

Consistency Checks: 100000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['4 ', '2 ', '7 ', '6 ', '3 ', '9 ', '8 ', '5 ', '1 ']

['3 ', '5 ', '0 ', '4 ', '8 ', '1 ', '0 ', '2 ', '7 ']

['8 ', '1 ', '0 ', '0 ', '5 ', '2 ', '9 ', '6 ', '4 ']

['1 ', '4 ', '3 ', '9 ', '0 ', '6 ', '2 ', '7 ', '5 ']

['2 ', '8 ', '5 ', '0 ', '1 ', '7 ', '4 ', '3 ', '6 ']

['6 ', '7 ', '0 ', '8 ', '4 ', '5 ', '1 ', '0 ', '9 ']

['0 ', '0 ', '6 ', '7 ', '2 ', '3 ', '5 ', '4 ', '0 ']

['5 ', '3 ', '9 ', '1 ', '6 ', '4 ', '7 ', '8 ', '2 ']

['7 ', '0 ', '2 ', '5 ', '9 ', '8 ', '6 ', '1 ', '3 ']

===================================================

root@ubuntu:/home/neeraj/Artificial\_Intelligence/arnav# python sudoku.py --input game.txt

Backtracking:

Execution Time: 60.359836

Consistency Checks: 4856444

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['8 ', '11 ', '12 ', '1 ', '7 ', '10 ', '5 ', '2 ', '6 ', '4 ', '3 ', '9 ']

['7 ', '10 ', '4 ', '2 ', '6 ', '8 ', '9 ', '3 ', '5 ', '12 ', '11 ', '1 ']

['3 ', '6 ', '9 ', '5 ', '1 ', '12 ', '4 ', '11 ', '7 ', '2 ', '10 ', '8 ']

['11 ', '4 ', '1 ', '12 ', '9 ', '5 ', '2 ', '10 ', '8 ', '7 ', '6 ', '3 ']

['6 ', '8 ', '5 ', '9 ', '4 ', '3 ', '7 ', '12 ', '10 ', '1 ', '2 ', '11 ']

['2 ', '3 ', '7 ', '10 ', '11 ', '6 ', '1 ', '8 ', '4 ', '9 ', '5 ', '12 ']

['5 ', '12 ', '8 ', '6 ', '10 ', '2 ', '11 ', '4 ', '1 ', '3 ', '9 ', '7 ']

['1 ', '9 ', '11 ', '3 ', '5 ', '7 ', '8 ', '6 ', '12 ', '10 ', '4 ', '2 ']

['10 ', '7 ', '2 ', '4 ', '12 ', '1 ', '3 ', '9 ', '11 ', '6 ', '8 ', '5 ']

['12 ', '5 ', '6 ', '8 ', '2 ', '9 ', '10 ', '7 ', '3 ', '11 ', '1 ', '4 ']

['9 ', '1 ', '10 ', '11 ', '3 ', '4 ', '12 ', '5 ', '2 ', '8 ', '7 ', '6 ']

['4 ', '2 ', '3 ', '7 ', '8 ', '11 ', '6 ', '1 ', '9 ', '5 ', '12 ', '10 ']

backtrackingMRV:

Execution Time: 15.867679

Consistency Checks: 1082876

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['8 ', '11 ', '12 ', '1 ', '7 ', '10 ', '5 ', '2 ', '6 ', '4 ', '3 ', '9 ']

['7 ', '10 ', '4 ', '2 ', '6 ', '8 ', '9 ', '3 ', '5 ', '12 ', '11 ', '1 ']

['3 ', '6 ', '9 ', '5 ', '1 ', '12 ', '4 ', '11 ', '7 ', '2 ', '10 ', '8 ']

['11 ', '4 ', '1 ', '12 ', '9 ', '5 ', '2 ', '10 ', '8 ', '7 ', '6 ', '3 ']

['6 ', '8 ', '5 ', '9 ', '4 ', '3 ', '7 ', '12 ', '10 ', '1 ', '2 ', '11 ']

['2 ', '3 ', '7 ', '10 ', '11 ', '6 ', '1 ', '8 ', '4 ', '9 ', '5 ', '12 ']

['5 ', '12 ', '8 ', '6 ', '10 ', '2 ', '11 ', '4 ', '1 ', '3 ', '9 ', '7 ']

['1 ', '9 ', '11 ', '3 ', '5 ', '7 ', '8 ', '6 ', '12 ', '10 ', '4 ', '2 ']

['10 ', '7 ', '2 ', '4 ', '12 ', '1 ', '3 ', '9 ', '11 ', '6 ', '8 ', '5 ']

['12 ', '5 ', '6 ', '8 ', '2 ', '9 ', '10 ', '7 ', '3 ', '11 ', '1 ', '4 ']

['9 ', '1 ', '10 ', '11 ', '3 ', '4 ', '12 ', '5 ', '2 ', '8 ', '7 ', '6 ']

['4 ', '2 ', '3 ', '7 ', '8 ', '11 ', '6 ', '1 ', '9 ', '5 ', '12 ', '10 ']

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['8 ', '11 ', '12 ', '1 ', '7 ', '10 ', '5 ', '2 ', '6 ', '4 ', '3 ', '9 ']

['7 ', '10 ', '4 ', '2 ', '6 ', '8 ', '9 ', '3 ', '5 ', '12 ', '11 ', '1 ']

['3 ', '6 ', '9 ', '5 ', '1 ', '12 ', '4 ', '11 ', '7 ', '2 ', '10 ', '8 ']

['11 ', '4 ', '1 ', '12 ', '9 ', '5 ', '2 ', '10 ', '8 ', '7 ', '6 ', '3 ']

['6 ', '8 ', '5 ', '9 ', '4 ', '3 ', '7 ', '12 ', '10 ', '1 ', '2 ', '11 ']

['2 ', '3 ', '7 ', '10 ', '11 ', '6 ', '1 ', '8 ', '4 ', '9 ', '5 ', '12 ']

['5 ', '12 ', '8 ', '6 ', '10 ', '2 ', '11 ', '4 ', '1 ', '3 ', '9 ', '7 ']

['1 ', '9 ', '11 ', '3 ', '5 ', '7 ', '8 ', '6 ', '12 ', '10 ', '4 ', '2 ']

['10 ', '7 ', '2 ', '4 ', '12 ', '1 ', '3 ', '9 ', '11 ', '6 ', '8 ', '5 ']

['12 ', '5 ', '6 ', '8 ', '2 ', '9 ', '10 ', '7 ', '3 ', '11 ', '1 ', '4 ']

['9 ', '1 ', '10 ', '11 ', '3 ', '4 ', '12 ', '5 ', '2 ', '8 ', '7 ', '6 ']

['4 ', '2 ', '3 ', '7 ', '8 ', '11 ', '6 ', '1 ', '9 ', '5 ', '12 ', '10 ']

backtrackingMRVfwd:

Execution Time: 4.967056

Consistency Checks: 370332

Solution printed above

backtrackingMRVcp:

Execution Time: 15.625522

Consistency Checks: 1082876

Solution:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['8 ', '11 ', '12 ', '1 ', '7 ', '10 ', '5 ', '2 ', '6 ', '4 ', '3 ', '9 ']

['7 ', '10 ', '4 ', '2 ', '6 ', '8 ', '9 ', '3 ', '5 ', '12 ', '11 ', '1 ']

['3 ', '6 ', '9 ', '5 ', '1 ', '12 ', '4 ', '11 ', '7 ', '2 ', '10 ', '8 ']

['11 ', '4 ', '1 ', '12 ', '9 ', '5 ', '2 ', '10 ', '8 ', '7 ', '6 ', '3 ']

['6 ', '8 ', '5 ', '9 ', '4 ', '3 ', '7 ', '12 ', '10 ', '1 ', '2 ', '11 ']

['2 ', '3 ', '7 ', '10 ', '11 ', '6 ', '1 ', '8 ', '4 ', '9 ', '5 ', '12 ']

['5 ', '12 ', '8 ', '6 ', '10 ', '2 ', '11 ', '4 ', '1 ', '3 ', '9 ', '7 ']

['1 ', '9 ', '11 ', '3 ', '5 ', '7 ', '8 ', '6 ', '12 ', '10 ', '4 ', '2 ']

['10 ', '7 ', '2 ', '4 ', '12 ', '1 ', '3 ', '9 ', '11 ', '6 ', '8 ', '5 ']

['12 ', '5 ', '6 ', '8 ', '2 ', '9 ', '10 ', '7 ', '3 ', '11 ', '1 ', '4 ']

['9 ', '1 ', '10 ', '11 ', '3 ', '4 ', '12 ', '5 ', '2 ', '8 ', '7 ', '6 ']

['4 ', '2 ', '3 ', '7 ', '8 ', '11 ', '6 ', '1 ', '9 ', '5 ', '12 ', '10 ']

minConflict:

Execution Time: 15.406985

Consistency Checks: 100000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

['1 ', '11 ', '0 ', '5 ', '7 ', '10 ', '2 ', '8 ', '6 ', '4 ', '3 ', '12 ']

['7 ', '10 ', '12 ', '2 ', '6 ', '4 ', '0 ', '3 ', '5 ', '1 ', '11 ', '8 ']

['4 ', '6 ', '9 ', '0 ', '1 ', '12 ', '5 ', '11 ', '7 ', '2 ', '10 ', '0 ']

['3 ', '4 ', '1 ', '7 ', '2 ', '5 ', '9 ', '10 ', '8 ', '12 ', '6 ', '11 ']

['11 ', '8 ', '5 ', '9 ', '4 ', '7 ', '6 ', '12 ', '10 ', '3 ', '2 ', '1 ']

['2 ', '0 ', '10 ', '6 ', '11 ', '3 ', '1 ', '0 ', '4 ', '9 ', '7 ', '5 ']

['9 ', '12 ', '8 ', '1 ', '10 ', '2 ', '3 ', '4 ', '11 ', '0 ', '5 ', '7 ']

['0 ', '2 ', '11 ', '3 ', '5 ', '6 ', '7 ', '1 ', '12 ', '10 ', '4 ', '9 ']

['5 ', '7 ', '0 ', '4 ', '12 ', '9 ', '11 ', '0 ', '1 ', '6 ', '8 ', '2 ']

['12 ', '5 ', '2 ', '8 ', '9 ', '0 ', '10 ', '7 ', '3 ', '11 ', '1 ', '0 ']

['10 ', '1 ', '0 ', '11 ', '3 ', '0 ', '4 ', '5 ', '2 ', '7 ', '9 ', '6 ']

['6 ', '9 ', '3 ', '0 ', '0 ', '1 ', '8 ', '2 ', '0 ', '5 ', '12 ', '4 ']

**References**

These links were referred for understanding the algorithms to solve sudoku

<http://www.cs.rochester.edu/~brown/242/assts/termprojs/Sudoku09.pdf>

[https://amoon.netfirms.com/Portfolio/Applied%20AI%20-%20Sudoku%20Solver.pdf](https://www.google.com/url?q=https%3A%2F%2Famoon.netfirms.com%2FPortfolio%2FApplied%2520AI%2520-%2520Sudoku%2520Solver.pdf&sa=D&sntz=1&usg=AFQjCNFlIZDYIdnz24pfCqwxPO5HG18EBg)