9 X 9 SUDOKU GAME

Team members:
Akash Raju Dasararaju
Saismaran Thanneru

PROBLEM DESCRIPTION:

Purpose: To solve the 9 X 9 sudoku game using backtracking search algorithm. It can be represented as a Constraint Satisfaction Problem (CSP), where:

X = {X1, ..., Xn}

Di = {1, 2, 3, 4, 5, 6, 7, 8, 9} is domain describing the values each variable can be assigned (Xi ∈ D),

C = "Each value appears once in each row, column and 3x3 block of the grid" is the set of constraints.

Four rules:

1. Allowed numbers are 1,2,3,4,5,6,7,8,9 only.
2. It is not allowed to repeat numbers in the same row.
3. It is not allowed to repeat numbers in the same column.
4. It is not allowed to repeat numbers in the 3 by 3 sub square grids.

APPROACH:

We'll define how the Backtracking algorithm works for this situation cell by cell & addressing the cells by the row and column. It is a recursive search, and we call it a depth first search because this search recursively considers all possibilities.

Methodology:

Backtracking algorithm is a technique for solving problems recursively by trying to build a solution incrementally one piece at a time, removing those solutions that fail to satisfy the constraints of the problem at any point of time. Backtracking is used when you need to find the correct series of choices that will solve a problem.

Conclusion:

The backtracking approach has been demonstrated to be superior to both the constraint and rule-based algorithms when comparing sudoku solution algorithms developed. The constraint solution was so inefficient that it should never be used for anything other than learning constraint programming . Even when merely counting the initiation and memory allocation, the rule-based approach was slower than backtracking, and as a result, adding more rules will never make it faster than backtracking

References:

Pseudocode (and working example) for the backtracking algorithm, Retrieved March 2015 Felgenhauer, Bertram; Jarvis, Frazer (June 20, 2005), Enumerating possible Sudoku grids http://ijcsi.org/papers/IJCSI1121247253.pdf Stanford lecture by Julie Zelensky with example of a backtracking algorithm https://www.youtube.com/watch?v=pgpaIGRCQI