Algorithms and Analysis Assignment 2 Report

Jialun Darren Huang (S3755729)

Implementation of Killer Advanced Solver - An enhancement from the Backtracker algorithm

The problem with backtracker is that since it takes in all available numbers in the sudoku list, whether it is correct or wrong logically.

One of the logic we will be looking to make use of is the cage.

	0	1	2	3	4	5	6	7	8
0	21				7		14	24	
1	24		7		28			5	
2			14						9
3	20			6		14	17		
4			17				13		23
5			23		17			14	
6	30					9			
7				18			19		
8			12						

Taken from killer_99.in

Let's put our focus on the cage that has the cells R0C0, R0C1, R0C2, R0C3 and R1C1. The total sum of the cage is 21, as shown on one of the numbers in the cage

	0	1	2	3	4	5	6	7	8
0	5	3	4	7					
1									
2									
3									
4									
5									
6									
7									
8									

Now, here is an example of the problem with backtracking. Say we know the cell values of R0C0, R0C1, R0C2 and R0C3. By logic, we can see that the R1C1 value would be 2 (21 - 5 - 3 - 4 - 7). However, since backtracking brute forces by putting all values from 1 to 9, we are wasting computational time. This means that we can make a more efficient algorithm by immediately putting 2 in R1C1 right now, than to wait for all R0C0 to R0C8 and R1C0 cells to be filled up.

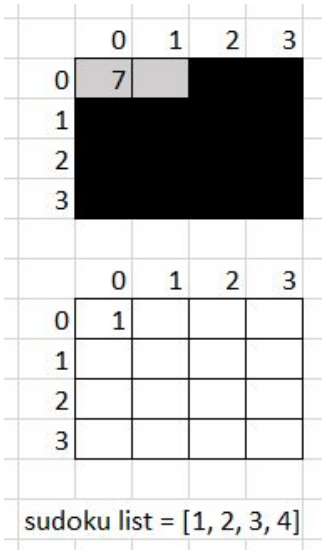
This also means that when we will have fewer puzzles to validate too as the constraint of R1C1 being 2 will affect all cells in Row 1 and Column 1.

Things to take note when implementing this algorithm

While this algorithm is definitely a breakthrough compared to the backtracking algorithm, extra space is required to determine whether this is possible.

Firstly, we have to determine whether this cage, where a number is added, is the second last cell in the cage. If this is true, this means that there is only a cell left that has no cell value in the cage.

Next, we have to know the value to be put in the cell, however, this value must exist in the sudoku list.



Here we can see that R0C1 is the last cell in the cage (R0C0 & R0C1). However, the value 6 (7 - 1) does not exist in the sudoku list. Therefore, R0C0 cannot be 1.

When backtracking, you have to make sure that all cages are being managed for, by setting the values of those cells in the cage to -1 (my default value to determine that a cell has no value). Otherwise, the last cell value will affect the validation (row, column and box constraints).