CS171 Project Progress report #2: Forward Checking

Name: Haoming Huang, Jason Chen-Ju Hsieh

Date: February 14, 2016

1 Scope

The purpose of this assignment is to implement the forward checking component of Sudoku solver. Our team will be continuing off of the provide Java shell and changes made from the last assignment.

2 Progress

We've refactored the code to return solver status. We moved the status variable to BTSolver with default value error and have it update depending on timeout or success.

We've also implemented forwarding checking based on existing functions. After the solver assigns a value to a variable, it first calls assignmentCheck() and checks that no neighboring variables (down, across, and within smaller boxes) are assigned to the same values. If not, return false so the solver can backtrack. Otherwise, remove the assigned value from the domains of all neighboring variables and return true.

We've looked ahead and changed the function to calculate time taken to consider preprocessing time, also added function to return preprocessing times of arc consistency.

3 Problems & Questions

It was difficult to understand how to use the tools already present to implement forward checking. It was unclear at first whether some already existing component(s) can be used or most of it is yet to be coded. But we quickly found that assignmentCheck() is necessary as part of forward checking and several function that manipulates the domains of variables can be used to implement the rest.

4 Results

The program takes in correct inputs (input file, output file, timeout, and FC token), correctly solves the Sudoku puzzle using the back track search implementation and uses the forwarding checking consistency check to remove inconsistent domain values to speed up the searching process, and outputs a file with the correct format. Back track search with FC implemented is sometimes twice as fast with less node counts