**SUDOKU SOLVER**

**Abstract**

Sudoku is a form of puzzle that is three rows and three columns of squares. Each square then holds another three by three set of boxes. This gives a board of 81 boxes that need to be filled with the numbers one through nine. Each row must contain the set from one to nine, each column must contain the set from one to nine, and each box must contain the set from one to nine. Certain boxes are already filled with a number to start the puzzle, and the solver must fill in the remaining numbers to complete the puzzle.

A simple Sudoku Solver was written using the Python programming language and is powerful enough to solve any proper Sudoku puzzle of any difficult.

**WORKING**

This program works on the basis of reverse backtrack algorithm. In this method we first looks for empty slot in Sudoku puzzle and test that slot with various numbers from 1 – 9. On adding one number to the slot the program checks for validity by searching for the same number in that column as well as row and also in the grid. If the number is not repeated it checks for another slot. If any of the number get invalid the programs return back to the earliest step and corrects that number.

If the puzzle is itself invalid it returns error. For graphics we are using PyQt5 and python 3.5 as basic language.

**Conclusion**

The Sudoku solver was fairly easy to implement and test. Though it is not very efficient, it will solve any valid puzzle correctly and in a small amount of time, and if the puzzle isn’t valid it will solve it as far as it can.