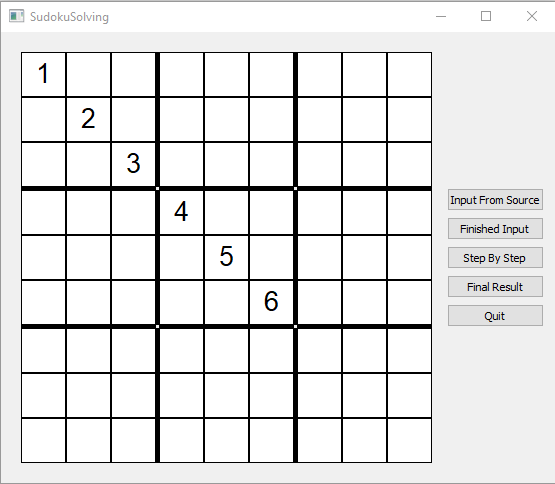
# Summary description

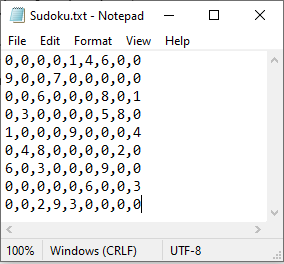
This is a simple application using QT framework for show my Sudoku solving, instead of using backtracking solution. This solution use human logical thinking (more detail at 2.2, 2.3, and 2.4).

This app is not a Sudoku game playing, so we need input question. We can set input from keyboard thought GUI or set input from text file.

Set input from keyboard though GUI:

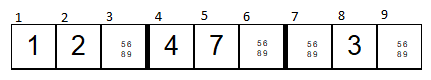


Set input form text file as the format below:



# Solution Description

## Main value and sub value

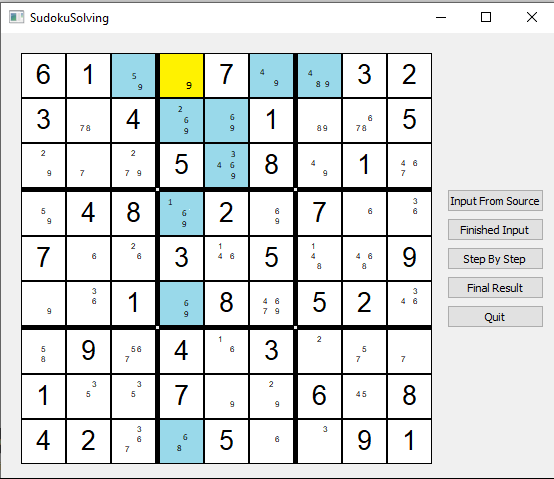


The above picture is a one row of Sudoku question. The corresponding value with first, second, fourth, fifth, eighth position is 1, 2, 4, 7, 3, I call them (1, 2, 4, 7, 3) is the “main value”. At the third position, it can receive 5 or 6 or 8 or 9, I call them is the “sub value”. The sub value is easy saving by bin, and convert to text for setText() function.

Ex: third position have sub value {5, 6, 8, 9} => 0b1 1011 0000 = 432

## First solution

For the first solution, it will find any position can receive only one value. As yellow position below.



After the yellow position is updated “main value”, any position has same row, or same column, or same square 3x3 with yellow position will be updated “sub value”.

EX: position[1,3] has sub value {5, 9} (0b1 0001 0000), it will be removed value 9:

0b1 0001 0000

& 0b0 1111 1111

= 0b0 0001 0000

EX: position[3,5] has sub value {3, 4, 6, 9} (0b1 0010 1100), it will be removed value 9:

0b1 0010 1100

& 0b0 1111 1111

= 0b0 0010 1100

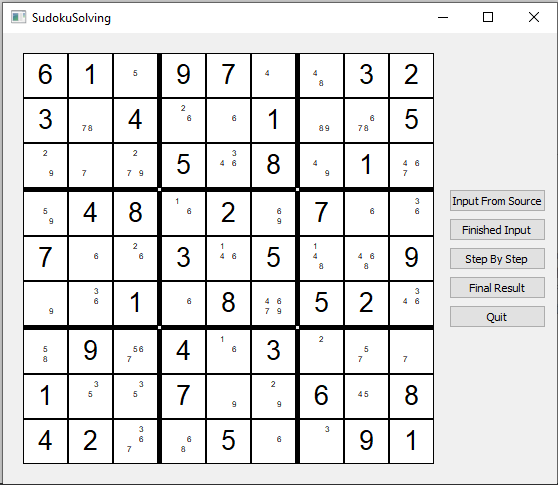
EX: position[1,7] has sub value {4, 8, 9} (0b1 1000 1000), it will be removed value 9:

0b1 1000 1000

& 0b0 1111 1111

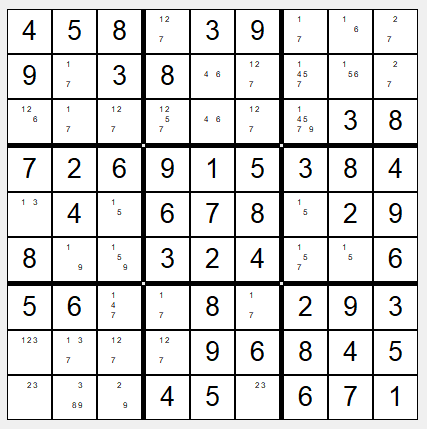
= 0b0 1000 1000

The above picture will be change to below picture:



## 2.3 Second solution

For the second solution, it will find only one value can be receive by only position of row, or column, or square 3x3. As below picture.



As you see on the first row, value 6 just be received by position[1,8], because there is no another position (only set for first row) can received value 6.

Algorithm is to find out:

position [1, 4] can receive {1, 2, 7} => sub value = 0b0 0100 0011 (A)

position [1, 7] can receive {1, 7} => sub value = 0b0 0100 0001 (B)

position [1, 8] can receive {1, 6} => sub value = 0b0 0010 0001 (C)

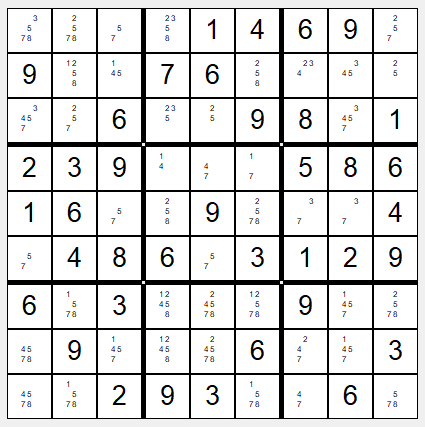
position [1, 9] can receive {2, 7} => sub value = 0b0 0100 0010 (D)

A | B | C | D = 0b0 0110 0011

A | B | D = 0b0 0100 0011

(A | B | C | D) XOR (A | B | D) = 0b0 0010 0000 (value is 6 for “C”)

## Third solution



As above picture, you see fifth row,

position [5, 3] can receive {5, 7}

position [5, 4] can receive {2, 5, 8}

position [5, 6] can receive {2, 5, 7, 8}

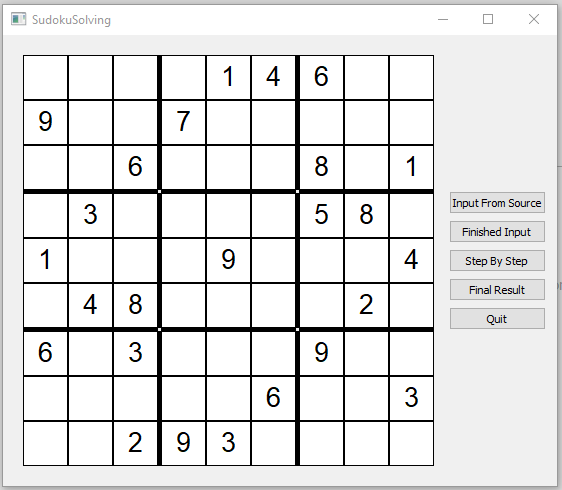
position [5, 7] can receive {3, 7}

position [5, 8] can receive {3, 7}

As you see if position[5,7] receive 3, position[5,8] will receive 7 and vice versa. So sub value of position[5,3], position[5,4], position[5,6] need to remove value 3, and 7.

After that, sub value of position[5,3] is only have value 5.

# GUI



As above picture, I define 5 button.

\* The “Input-From-Source” button is used for set input from text file. It will show QFileDialog for get text file.

You can set input from keyboard though GUI.

\* Whenever finishing the setting input, you click the “Finished Input” button, sub value of all position will be shown.

\* The ”Step By Step” button will help you solving question step by step.

\* The “Final Result” button will show final result.

\* The “Quit” button is used to close application.