

# CSE 6010 / CX 4010 Assignment 4

## Logic Puzzle

Ting Liao, GTID: 903278773 | Chin Wang, GTID: 903506612

09/21/2020

For this Logic Puzzle assignment, we tried to use the idea similar as divide and conquer. The main reason is that we tried to assign the division of labor across the team evenly. Also, It is much easier for us to focus on a small part without being distracted by other code. What's more, due to that comparing characters is complicated, so we decided to transform the puzzle array from character to integer. The purpose of each function, for-loop enable us to loop through the item in the puzzle and to do the calculation or comparing with other items in order to satisfy the three rules.

# Description of our Approach

- At first, we tried to use Rule-1 and Rule-2 functions to solve the puzzle and found out that puzzle with 6x6 could be solved by only using Rule-1 and Rule-2 functions iteratively. So we thought that using only Rule-1 and Rule-2 was efficient for 10x10 and 12x12 puzzles. However, when we ran the program with 10x10 puzzle, the result still had some '?'s in the puzzle. Therefore, we needed to figure out whether any other rule could be applied in this program.
- We started solving the puzzles manually and realized that another method rather than Rule-1, Rule-2 could be used. When any row or column had only one 'O' or 'X' to fill in the '?', some '?'s could be solved and then we could continue to solve the remaining '?'s. Nevertheless, this method involved too many boundary conditions and it would be too complicated to be constructed. In other words, there should be a way for computer to solve rather than human being. In computer, recursive method could efficiently be performed if the algorithm was developed properly. At last, we found out binary tree method might work.

# Future improvement

- In this program, we converted '?', 'X', 'O' into 0, 1, 2. We think that there might be a way to use original data to implement the program, so we might adjust the program in the future to avoid any unnecessary transformation of the puzzle.
- Also, It would be better if we could integrate some for-loops or small functions together to make our code much cleaner and clearer.
- Due to that we use a lot of nested for-loops, the time complexity is pretty bad. Some of the function need to run  $O(n^3)$ . We should find another algorithm method or data structure to lower down the time complexity.

# Evidence of Correct Operation

- From the figures shown below, as you can see in figure 1, we successfully read in our puzzle array and transform the character to integer. Then, we start to do the try-assign function. Finally, after several times of recursion, we got our final result which shows that our program can successfully solve the puzzle without violating any rule.
- In figure 2, it shows the process of how we assign "X" or "O" into the puzzle array in each function. In figure (c), you can see that in the third and the fifth line, when the program found that assigning "X" would fail, it would change to assign "O" in that place.

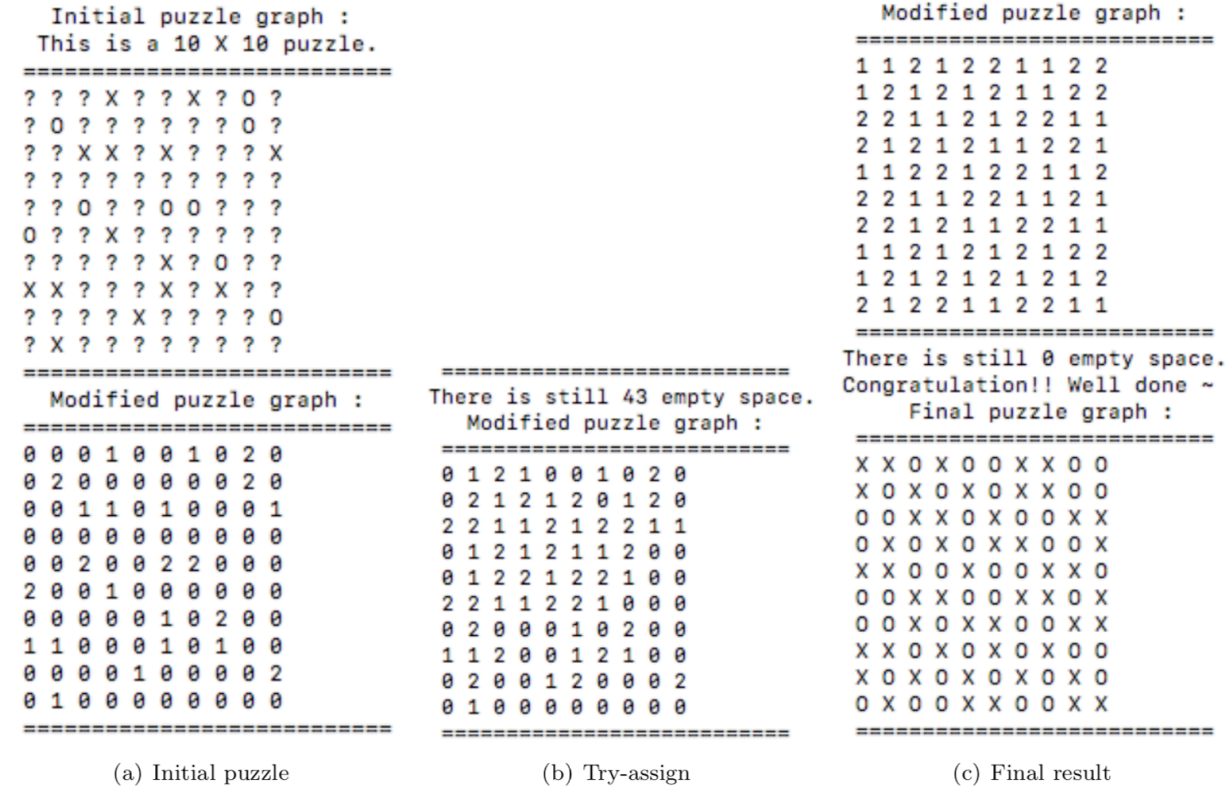


Figure 1: Puzzle program correction evidence - Puzzle Graph

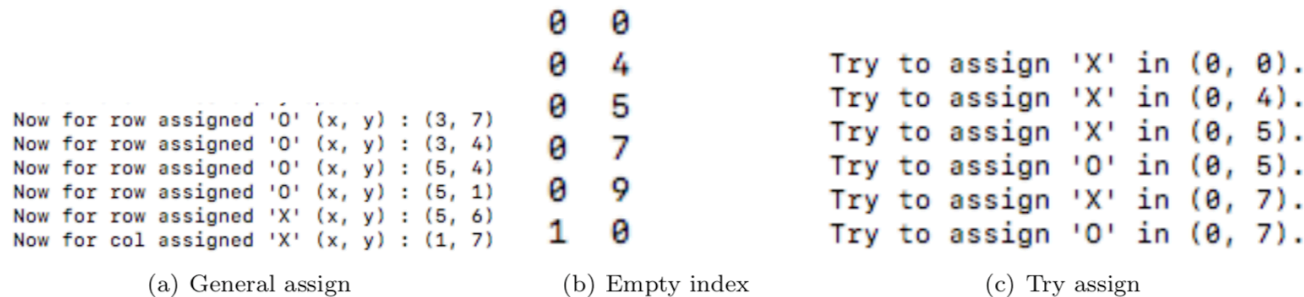


Figure 2: Puzzle program correction evidence - Executing Process

# Division of Labor

- For the small scale function:
  - Ting Liao:  
function -> get\_xo\_num\_row, get\_xo\_num\_col, get\_Q\_index, check\_row\_col\_valid, check\_rule\_1and2,
  - Chin Wang:  
function -> check\_consecutive, check\_jump, check\_sameNum\_XO, check\_rule\_3, readIn\_file
- For the Integration part:
  - Try\_assign\_random function:
    - The code is typed by Ting Liao
    - The idea of using recursion is from Chin Wang. Chin came up with the idea because of solving Sudoku.
  - Main function:
    - The readIn file was written by Chin Wang
    - The rest of the code was modified several times by both of us
- Basically, we have read and tried to understand each other's code and gave some suggestions on it. Also, almost all the functions have been modified by the other one. Therefore, the above division of labor is just a rough division and both of us all put great effort on completing this assignment.