Problem Statement

***Grade: 10 / 10***

***Good Job! It will be nice if you put the team name and all team members names as the subtitle of this document.***

1. Newspaper puzzle solvers. For example a sudoku solver and/or calcudoku solver.

2a. The software would solve the blank squares in sudoku. Some issues that can arise are guesses between two or more numbers that need to be made if there is no clear path and going back if the guess was wrong. Connecting the columns, rows, and squares with solved and unsolved tiles. The interface with the user should look like a board and the user should be able to input the numbers and the computer should solve at least 1 possibility of the board. *2 / 10*

2b. We chose to make a Sudoku solving program as our semester project because it will be a challenge, yet it is still attainable. Along with Sudoku, we are also giving ourselves the option to expand it to other variations of Sudoku, which gives the software adaptability. We expect this project to help us improve our algorithm creating and discrete mathematics skills. Another large part of our program will be creating a simple graphic user interface that is easy to understand for various ages and skill levels. Sudoku is a game known on a global scale that utilizes the universal language of mathematics. This makes our potential program a useful tool for people all over the world that are stuck on that difficult puzzle. *2 / 10*

2c. The guaranteed feature of the program will be a 9x9 Sudoku puzzle solver. The user will be able to input the given values into the program via a GUI, and the program will process the data and return a completed puzzle. If this is completed early, additional features will include a size magnifier, allowing for any size puzzle to be solved, provided it is square, a shape changer, to allow for different configurations of puzzles to be solved, and, possibly, solvers for other “newspaper puzzles;” this could include Kakure puzzles, word searches, calcudokus, jumbles, and other newspaper puzzles. *3 / 10*