Justin Grubbs Final Project:

Sudoku

Overview:

* Overall, I am fairly pleased with how my project ended up. I had already slightly started my project when I created my design document, so the general design stayed fairly true; however, I ran into many unanticipated issues that required revising
* True to my design document, I ended up using an enum for the difficulties, multiple HashMaps to organize all of the data, buttons to control the different highlighting options, and file input to select the grid and respective key
* I ended up opting to not build the puzzle generator, as I ran into several unforeseen issues that left me with not enough time

Unique and Interesting Features:

* My favorite features are the options buttons
  + Although they proved difficult to implement, I think it was well worth the effort as it helps with the clarity of the puzzle and adds a bit of color, all the while being able to be disabled and reenabled as the user sees fit
* I also enjoy the ability to navigate around the grid using the arrow keys
* Being able to use a single index to access the current grid, answer key, specific cell, and specific label proved very useful and likely saved me a lot of time and headache
* I used six different HashMaps to organize my data, nearly all of which have keys ranging from 1 to 81
  + currentKey:
    - Value: int ranging from 0 to 9, depending on the current state of the cell/grid
  + answerKey:
    - Value: int ranging from 1 to 9, depending on whatever the correct value of the cell is
  + intersectionKey:
    - Value: array representing the cells that intersect the index
  + matchingCellKey:
    - Key: int ranging from 1 to 9, representing the different possible values
    - Value: ArrayList containing the indices of all the cells containing the same value as the key
  + cellMap and labelMap:
    - Value: contains all cells and all labels respectively

Joys and Struggles:

* The main issues I encountered with this project were caused by the option buttons and the different highlighting features. In the zip file there is a previous version of my controller that was riddled with mistakes. I would “fix” something only to find that it created two more problems. Furthermore, the code was very bloated and all over the place. I ended up just walking through it and removing much of the random quick fixes, which solved all of the highlighting issues immediately
* In the lab that we worked on this project, I had made a view for the cells but decided to scrap it as it would make me redo the intersecting highlights that I had already made. If I were to redo this project, I would most definitely implement at least one view so that my code is less overwhelming and confusing