Kyle Morgan

Professor Tao

CSC 360, Sec. 03

22 February 2023

**Tic-Tac-Toe Analysis**

Starting off with the analysis of the structure. The file’s main class is at the very end and just runs the command launch(args); .“TicTacToeWithSaveGame” which is already an extension of the class “Application” is the largest class. The main class consists of several functions that do a variety of functions such as initializing the board, prompting players, and event controls game save. Along with the main class there is an additional class called “Cell”. “Cell” is an extension of the class called Pane and is called on a lot in the class “TicTacToeWithSaveGame”. The “Cell” class handles tokens and handles mouse click events to check if won.

This code utilizes exception handling which is a topic that we started to learn about early on in the class. This allows the program to catch an error and try something to fix the issue. The program also seems to use I/O which we have learned about and handling files. Otherwise this program uses typical Object-Oriented programming techniques that we have made standard. Especially involving the format of the code making it easier to read.

The structure of the code is implemented via a 2D array for the game board. The save game functionality is saved to a text file and the game state is saved as a string representation of the game board. To load a game the file is read.

Some limitations of this program are but not limited to: The game doesn’t seem to prevent players from overwriting each other. On the topic of overwriting it seems that with every save the previous game will also get overwritten.