<u>Objects and their responsibilities:</u>

MainGame: This is the primary controller for the game, and its main responsibility is to manage the game's state and logic. It switches between guess and answer modes, checks if the game has been won, plays a sound when the game is won, and initializes the game board using the SudokuInitializer.

Cell: This object represents an individual cell on the Sudoku board. Its main responsibility is to store the value of the cell (if it has been filled in), as well as any guesses that have been made for this cell. It also checks if the value or guesses for this cell are valid according to the rules of Sudoku.

GUI: This is the graphical user interface for the game. Its main responsibility is to display the game to the user and handle user inputs. It draws the game and individual cells, and updates the game state when the user clicks on a cell or enters a guess or value.

SoundEffect: This object represents a sound effect that can be played in the game. Its main responsibility is to play a sound file, which is used by the MainGame to play a sound when the game is won.

SudokuInitializer: This object's main responsibility is to generate the initial state of the Sudoku board. It has a method to generate a 2D integer array representing a Sudoku board, which is used by the MainGame to initialize the game. This could involve reading from a file, generating a random valid Sudoku puzzle, or some other method of creating a Sudoku board.

SCROLL DOWN FOR SUDOKU DESIGN PLAN

MainGame

    Attributes:

        gameGrid: Cell[][]

        isGameWon: boolean

        isGuessMode: boolean

        initializer: SudokuInitializer

    Methods:

        switchMode(): void

        checkWin(): void

        playSound(): void

        initializeBoard(): void

Cell

    Attributes:

        x: int

        y: int

        value: int

        guesses: TextBox[4]

        answer: TextBox

    Methods:

        setValue(value: int): void

        addGuess(guess: int): void

        removeGuess(guess: int): void

        checkError(): boolean

GUI

    Attributes:

        game: MainGame

    Methods:

        drawGame(): void

        drawCell(cell: Cell): void

        clickCell(x: int, y: int): void

        enterGuess(x: int, y: int, guess: int): void

        enterValue(x: int, y: int, value: int): void

SoundEffect

    Attributes:

        soundFile: string

    Methods:

        play(): void

SudokuInitializer

    Attributes:

        None

    Methods:

        generateBoard(): int[][]

Relationships

MainGame and Cell: This is a Composition relationship. The MainGame class is composed of Cell objects, representing the Sudoku board. If the MainGame object is destroyed, so are the Cell objects.

MainGame and SudokuInitializer: This is a Dependency relationship. The MainGame class depends on the SudokuInitializer class to create the initial board layout.

MainGame and SoundEffect: This is also a Dependency relationship. The MainGame class depends on the SoundEffect class to play a sound when the game is won.

GUI and MainGame: This is an Association relationship. The GUI class is associated with the MainGame class, as it needs to interact with MainGame to update the game state based on user input and to display the current game state to the user.

Description of Classes and Methods

MainGame: This class manages the overall game operations.

    gameGrid: A 2D array of Cell objects that represent the Sudoku game grid.
    isGameWon: A boolean flag indicating whether the game has been won.
    isGuessMode: A boolean flag indicating whether the game is in guess mode or answer mode.
    initializer: An instance of the SudokuInitializer class used to generate the initial game board.
    switchMode(): Switches the game between guess mode and answer mode.
    checkWin(): Checks if game has been won (i.e., if Sudoku grid is completely and correctly filled).
    playSound(): Plays a sound effect when the game is won.
    initializeBoard(): Uses the SudokuInitializer to populate gameGrid with the initial game board.

Cell: This class represents a single cell on the Sudoku grid.

    x and y: The coordinates of the cell on the grid.
    value: The value of the cell, if it has been filled in.
    guesses: An array of TextBox objects for the guesses placed on this cell (up to 4).
    answer: A TextBox object for the answer placed on this cell.
    setValue(): Sets the value of the cell.
    addGuess(): Adds a guess to the cell.
    removeGuess(): Removes a guess from the cell.
    checkError(): Checks if the value or guesses for the cell are valid according to rules of Sudoku.

GUI: This class handles the user interface and user interactions.

    game: An instance of the MainGame class to control the game state.
    drawGame(): Draws the entire game grid on the screen.
    drawCell(): Draws an individual cell on the screen.
    clickCell(): Handles the event of a cell being clicked.
    enterGuess(): Handles the event of a guess being entered.
    enterValue(): Handles the event of an answer being entered.

SoundEffect: This class handles playing sound effects.

    soundFile: The file path of the sound to be played.
    play(): Plays the sound effect.

SudokuInitializer: This class generates the initial state of the Sudoku board.

    generateBoard(): Returns a 2D integer array that represents a Sudoku board. This could involve reading from a file, using a generation algorithm, or other methods depending on the requirements.