23-813-0207: Lab 4 - Java Programming Lab

Lab exercise - 2

Date: 8/2/2024

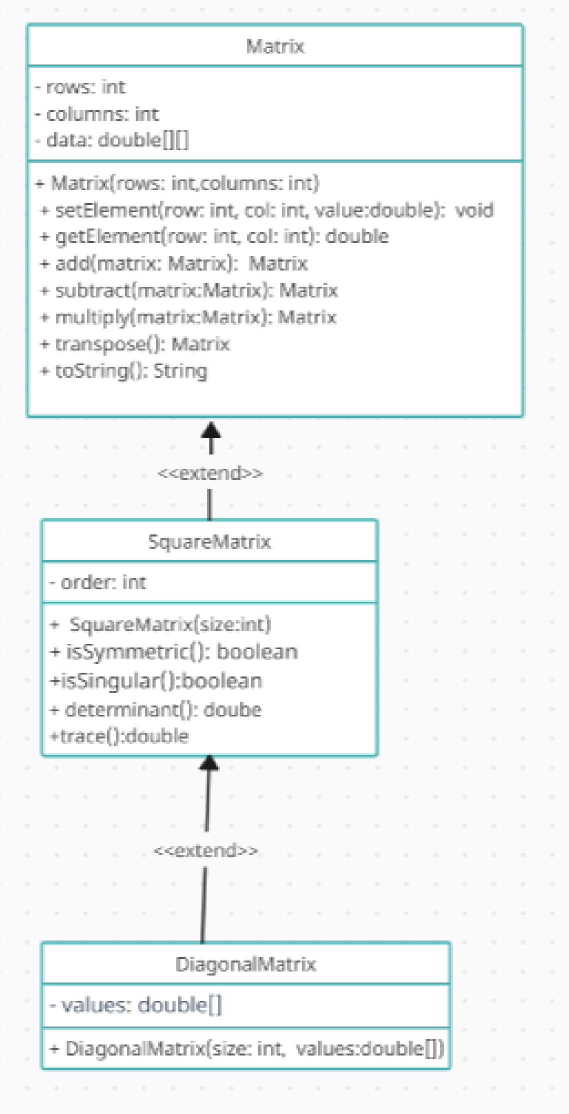

Fig 1

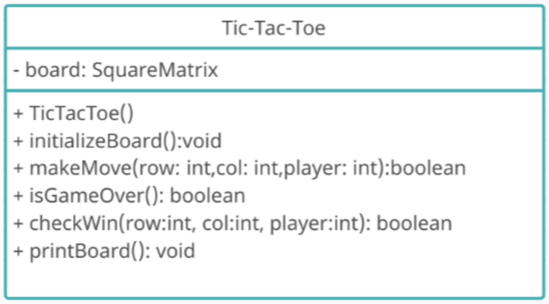

Fig 2

1. Implement the subclasses SquareMatrix and Diagonal Matrix as shown in the UML diagram above (Fig 1). Demonstrate all methods with suitable test cases.
2. Write a Java program to implement the tic-tac-toe game, as shown in the UML design above (Fig 2). The program should alternatively read row and column numbers from the two users and check if the move is valid. If the move is invalid, alert the user and ask for

a new move. If a valid move, update the board, print it, and check for a winner. If no winner yet, check if the game is over. Otherwise, continue.

The explanation of each method is given below.

- `board`: A private instance variable representing the game board as a `SquareMatrix`.
- `TicTacToe()`: Constructor for initializing a new game of Tic-Tac-Toe.
- `initializeBoard()`: Method to initialize the game board with all zeroes.
- `makeMove(row: int, col: int, player: int): boolean`: Method to make a move on the game board for the specified player (-1 for player X, 1 for player 0). Returns true if the move is valid, false otherwise.
- `isGameOver(): boolean`: Method to check if the game is over (either a player wins or the board is full). If there are no zeroes, board is full.
- `checkWinner(row: int, col: int, player: char):`boolean : Method to check the player has won after the new move.
- `printBoard(): void`: Method to print the current state of the game board. Remember to use X and O (not 1 and -1) while printing the board.