

Name: Shreya Panikkassery  
Roll no: 9562  
TE Comps A

AI Experiment no. 1

Code:

```
import java.util.Scanner;

public class TicTacToeBruteForce {

    private static int[][] moveTable = new int[(int) Math.pow(3, 9)][9];

    private static void initializeMoveTable() {
        for (int i = 0; i < Math.pow(3, 9); i++) {
            int[] ternary = convertToTernary(i);
            for (int j = 0; j < 9; j++) {
                moveTable[i][j] = ternary[j];
            }
        }
    }

    private static int[] convertToTernary(int decimal) {
        int[] ternary = new int[9];
        for (int i = 8; i >= 0; i--) {
            ternary[i] = decimal % 3;
            decimal /= 3;
        }
        return ternary;
    }

    private static int convertToDecimal(int[] ternary) {
        int decimal = 0;
        for (int i = 0; i < 9; i++) {
            decimal = decimal * 3 + ternary[i];
        }
        return decimal;
    }

    private static void printBoard(int[] board) {
        System.out.println("-----");
        for (int i = 0; i < 3; i++) {
            System.out.print("| ");
            for (int j = 0; j < 3; j++) {
                int index = i * 3 + j;
                char symbol = (board[index] == 1) ? 'X' : ((board[index] == 2) ? 'O' : ' ');
                System.out.print(symbol + " | ");
            }
        }
    }
}
```

```

    }
    System.out.println();
    System.out.println("-----");
}
}

private static boolean isValidMove(int[] board, int row, int col) {
    int index = row * 3 + col;
    return board[index] == 0;
}

private static void playerMove(int[] board, Scanner scanner) {
    int row, col;
    do {
        System.out.println("Enter your move (row and column, separated by space): ");
        row = scanner.nextInt() - 1;
        col = scanner.nextInt() - 1;
    } while (!isValidMove(board, row, col));

    int index = row * 3 + col;
    board[index] = 1; // Player move ('X')
}

private static void computerMove(int[] board) {
    int decimal = convertToDecimal(board);
    int[] moveVector = moveTable[decimal];

    for (int i = 0; i < 9; i++) {
        if (moveVector[i] == 0) {
            board[i] = 2; // Computer move ('O')
            return;
        }
    }
}

private static boolean isGameOver(int[] board) {
    return checkWin(board, 1) || checkWin(board, 2) || isBoardFull(board);
}

private static boolean checkWin(int[] board, int symbol) {
    // Check rows, columns, and diagonals for a win
    for (int i = 0; i < 3; i++) {
        if (board[i * 3] == symbol && board[i * 3 + 1] == symbol && board[i * 3 + 2] ==
symbol ||
            board[i] == symbol && board[i + 3] == symbol && board[i + 6] == symbol) {
            return true;
        }
    }
}

```

```

    }
}

if (board[0] == symbol && board[4] == symbol && board[8] == symbol ||
    board[2] == symbol && board[4] == symbol && board[6] == symbol) {
    return true;
}

return false;
}

private static boolean isBoardFull(int[] board) {
    for (int cell : board) {
        if (cell == 0) {
            return false; // Board is not full
        }
    }
    return true; // Board is full (draw)
}

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    initializeMoveTable();

    int[] board = new int[9];
    int currentPlayer = 1; // Player always plays as 'X'

    while (!isGameOver(board)) {
        printBoard(board);

        if (currentPlayer == 1) {
            playerMove(board, scanner);
        } else {
            computerMove(board);
        }

        currentPlayer = 3 - currentPlayer; // Switch player (1 to 2 or 2 to 1)
    }

    printBoard(board);
    if (checkWin(board, 1)) {
        System.out.println("Congratulations! You win!");
    } else if (checkWin(board, 2)) {

```

```

        System.out.println("Computer wins! Better luck next time.");
    } else {
        System.out.println("It's a draw! The game is over.");
    }

    scanner.close();
}
}

```

Output:

```

1 2
-----
| o | x |  |
-----
|  |  | x |
-----
|  |  |  |
-----
| o | x | o |
-----
|  |  | x |
-----
|  |  |  |
-----
Enter your move (row and column, separated by space):
2
2
-----
| o | x | o |
-----
|  | x | x |
-----
|  |  |  |
-----
| o | x | o |
-----
| o | x | x |
-----
|  |  |  |
-----
Enter your move (row and column, separated by space):
3
2
-----
| o | x | o |
-----
| o | x | x |
-----
|  | x |  |
-----
Congratulations! You win!
PS C:\Users\Windows\Desktop\java>

```