

## CCDSTRU

### Machine Project

#### Members:

Kimberly Ysabel S. Alejo  
Aaron Mark Armas  
Vivian Angel Chen

#### Mechanics of the game:

The game begins with an empty 6x6 board which is divided into 4 parts (3x3 boards). The players can choose where to place their markers with player 1 going first. The player is required to place their marker on any empty space in the board. After every move a player makes, the code checks for winning conditions (explained below) and declares the winner. If nobody wins it would be counted as a tie.

#### Winning Conditions:

1. A player wins if they are able to place markers on all of the following ordered pairs in TWO quadrants:
  - a. Pair One {(1, 2), (2, 1)}
    - i. Quadrant 1: {(1, 5), (2, 4), (2, 5), (2, 6), (3, 5)}
    - ii. Quadrant 3: {(4, 1), (4, 2), (4, 3), (6, 1), (6, 2), (6, 3)}
  - b. Pair Two {(1, 1), (2, 2)}
    - i. Quadrant 2: {(1, 1), (1, 3), (2, 2), (3, 1), (3, 3)}
    - ii. Quadrant 4: {(4, 4), (4, 6), (5, 5), (6, 4), (6, 6)}
2. Example:

| (2x2 square) | 1     | 2     |
|--------------|-------|-------|
| 1            | (1,1) | (1,2) |
| 2            | (2,1) | (2,2) |

| (6x6 square) | Quadrant 2 |       |       | Quadrant 1 |       |       |
|--------------|------------|-------|-------|------------|-------|-------|
| 1            | (1,1)      | (1,2) | (1,3) | (1,4)      | (1,5) | (1,6) |
| 2            | (2,1)      | (2,2) | (2,3) | (2,4)      | (2,5) | (2,6) |
| 3            | (3,1)      | (3,2) | (3,3) | (3,4)      | (3,5) | (3,6) |
|              | Quadrant 3 |       |       | Quadrant 4 |       |       |

|   |       |       |       |       |       |       |
|---|-------|-------|-------|-------|-------|-------|
| 4 | (4,1) | (4,2) | (4,3) | (4,4) | (4,5) | (4,6) |
| 5 | (5,1) | (5,2) | (5,3) | (5,4) | (5,5) | (5,6) |
| 6 | (6,1) | (6,2) | (6,3) | (6,4) | (6,5) | (6,6) |
|   | 1     | 2     | 3     | 4     | 5     | 6     |

3. Only one player can win. If all the spaces are filled but neither player were able to meet the winning conditions above its considered a tie.

### How The Code Works:

breakdown of the **main** function:

1. Initialization:
  - a. Prompts the user to specify which player ('X' or 'O') goes first
  - b. Prints the initial empty grid.
2. Game Loop:
  - a. Continues until the game is over.
  - b. Inside the loop:
  - c. Prompts the current player to input their move (row and column).
  - d. Checks the validity of the input move (within grid bounds, not occupied).
  - e. Updates the grid with the player's move.
  - f. Prints the updated grid.
  - g. Checks for win conditions:
  - h. checkWinXback: Checks for a backward slash win pattern.
  - i. checkWincross\_equalSlash: Checks for a cross or equal slash win pattern.
  - j. If either player wins, the game ends.
  - k. Checks for a tie condition if all spaces are filled.
  - l. Switches the player for the next turn.
3. **Functions:**
  - a. printGrid:
    - i. Takes a 2D array grid as input
    - ii. Copies the contents of the grid into a temporary array temp.
    - iii. Prints the grid with row and column numbering.
  - b. checkInput:
    - i. Validates the user's input move.
  - c. checkAvailability:
    - i. Checks if the specified grid position is available
  - d. checkWinXback:
    - i. Checks for a win condition with a backward slash pattern.
  - e. checkWincross\_equalSlash:
    - i. Checks for a win condition with a cross or equal slash pattern.