# **Assignment 2: Detecting a Result**

# Objective

In this assignment, you will implement and test functions that determine the game's outcome. You will extract specific board sequences, check for winning streaks, and detect ties when the board is full.

# Instructions

- 1. Complete each function according to the specifications described below.
- 2. After implementing a function, run tests.py to confirm that it works as specified. Many functions depend on previous ones, so complete them in order. Do not move on to the next function until the current one works correctly.
- 3. If a test fails, fix the related function and re-run your tests.
- 4. The tests.py file includes test cases for some functions. However, some functions do not have tests yet. You will need to write your own tests for those functions.

# **Tips for Testing:**

- Make multiple test calls to each function. Consider edge cases and potential failures.
- Test edge cases, such as:
  - Extracting sequences at the edges or corners of the board.
  - Checking for streaks of different lengths.
  - Handling full and nearly full boards.
- For functions that return Boolean values, test both true and false cases.

# **Function Descriptions**

### get\_row(board, row)

This function retrieves all values from a specified row and returns them as a list. The values should be in **left-to-right** order, as they appear on the board.

# get\_column(board, column)

This function retrieves all values from a specified column and returns them as a list. The values should be in **top-to-bottom** order, as they appear on the board.

#### get\_left\_diagonal(board, row, column)

This function extracts the **left diagonal** (top-left to bottom-right) which contains the value at the given row and column.

### get\_right\_diagonal(board, row, column)

This function extracts the **left diagonal** (top-left to bottom-right) which contains the value at the given row and column.

#### has\_streak(sequence, token, length)

This function checks if a sequence (row, column, or diagonal) contains a consecutive streak of a player's tokens.

- Loop through the sequence and count consecutive occurrences of token.
- If the count reaches the specified length, return True.
- If the sequence breaks (i.e., a different token or an empty space appears), reset the count.
- If no streak of the required length is found, return False.

#### check\_win(board, row, column, length)

After a player places a token, this function determines if they have won the game.

- Extract the **row**, **column**, **left diagonal**, and **right diagonal** sequences that pass through the given position.
- Use has\_streak() to check if any sequence contains a streak of at least length.
- If any sequence contains the required streak, return True (the player wins).
- Otherwise, return False.

### board\_full(board)

This function checks if the entire board is full.

- Loop through the board and check for any remaining empty spaces (θ).
- If all slots are occupied, return True (indicating a tie).
- Otherwise, return False.