Part I

Mathematical function:

1. Center position bonus
    - AI is the AI player
    - Opp is the opponent
    - Using 0-based indexing so that column 0 is the first column
    - If we have the M x N board (B) and p is my AI's piece (player 1 or player 2), then the center bonus can be represented as:

$$M_{center} = \{i \mid B_{i,\, floor(M/2)} = p\}$$

$$S_{center} = |\, 6 * M_{center}|$$

2. Window score:
    - Let w be the set of all such 4-cell windows on the board. For any window $w \in W$:
    - $aiCount(w)$ = number of cells in the window that are equal to p (your player)
    - $oppCount(w)$ = number of cells in the window that belong to your opponent
    - $emptyCount(w)$ = the number of empty cells in the window
    - Essentially we can define a piecewise function F(w) using the previous values:

$$F(w) = \begin{cases} +1000 & \text{if } aiCount(w) = 4 \\ +15 & \text{if } aiCount(w) = 3 \text{ and } emptyCount(w) = 1 \\ +6 & \text{if } aiCount(w) = 2 \text{ and } emptyCount(w) = 2 \\ +25 & \text{if } aiCount(3) \text{ and } w_1 = 0 \text{ and } w_4 = 0 \\ -18 & \text{if } oppCount(w) = 3 \text{ and } emptyCount(w) = 1 \\ -5 & \text{if } oppCount(w) = 2 \text{ and } emptyCount(w) = 2 \\ 0 & \text{otherwise} \end{cases}$$

    - Then we can get the total window score with the following function:

$$Window\ Score = \sum_{w \in W} F(w)$$

3. Final board score:

$$Board\ Score = S_{center} + Window\ Score$$

$$= |6 * M_{center}| + \sum_{w \in W} F(w)$$

Worked example:

initial board

			1			
		2	1			
		2	2	1		
2	1	1	2	2	2	
1	2	1	1	2	1	

= opp
= AI

step 1:center column bonus

$$center = \lfloor 7/2 \rfloor = 3$$

$$column\ 3 = [B_{0,3}, B_{1,3}, B_{2,3}, B_{3,3}, B_{4,3}, B_{5,3}]$$

$$= [0, 1, 1, 2, 2, 1]$$

$$\#\ of\ p\ in\ column: 1 + 1 + 1 = 3$$

$$S_{center} = 3 \times 6 = 18$$

step 2: evaluating horizontal windows

row 0 :

possible horizontal windows :

[0000], [0000], [0000], [0000]

windows meeting conditions : none so score = 0

row 1 :

possible horizontal windows :

[0001], [0010], [0100], [1000]

windows meeting conditions : none so score = 0

row 2 :

possible horizontal windows :

[0021], [0210], [2100], [1000]

windows meeting conditions : none so score = 0

row 3 :

possible horizontal windows :

[0022], [0221], [2210], [2100]

windows meeting conditions :

[0022] ⟶ score -= 5 ⟶ score = -5

row 4 :

possible horizontal windows :

[2112], [1122], [1222], [2220]

windows meeting conditions :

[2220] ⟶ score -= 18 ⟶ score = -23

row 5 :

possible horizontal windows :

[1211], [2112], [1121], [1210]

windows meeting conditions : none so score = -23

step 3: evaluating vertical windows

column 0 :

possible vertical windows :

[0000], [0002], [0021]

windows meeting conditions : none so score = -23

column 1 :

possible vertical windows :

[0000], [0001], [0012]

windows meeting conditions : none so score = -23

column 2:

possible vertical windows:

[0022], [0221], [2211]

windows meeting conditions:

[0022] ⟶ score -= 5 ⟶ score : -28

column 3:

possible vertical windows:

[0112], [1122], [1221]

windows meeting conditions: none so score = -28

column 4:

possible vertical windows:

[0001], [0012], [0122]

windows meeting conditions: none so score = -28

column 5:

possible vertical windows:

[0000], [0002], [0021]

windows meeting conditions: none so score = -28

column 5 :

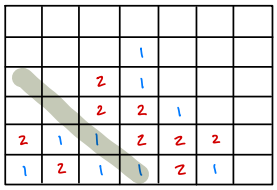
possible vertical windows :

[0000], [0000], [000]

windows meeting conditions : none so score = -28

step 4: evaluating diagonal down right windows

diagonal$_0$:



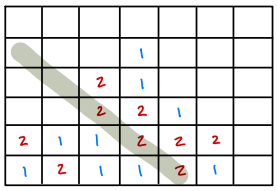
possible down right windows :

[0011]

windows meeting conditions :

[0011] ⟶ score += 6 ⟶ score = -22

diagonal$_1$:



possible down right windows :

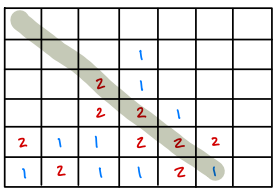
[0022], [0222]

windows meeting conditions :

[0022] ⟶ score -= 5 ⟶ score = -27

[0222] ⟶ score -= 18 ⟶ score = -45

diagonal$_2$:



possible down right windows :

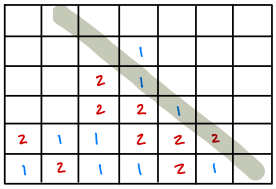
[0022], [0222], [2221]

windows meeting conditions :

[0022] ⟶ score -= 5 ⟶ score = -50

[0222] ⟶ score -= 18 ⟶ score = -68

diagonal $_3$:



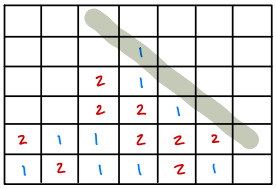
possible down right windows :

[0011], [0112], [1120]

windows meeting conditions :

[0011] ⟶ score += 6 ⟶ score = -62

diagonal $_4$:

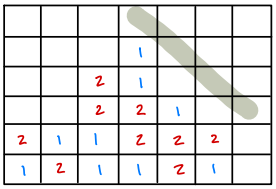


possible down right windows :

[0100], [1000]

windows meeting conditions : none so score = -62

diagonal $_5$:



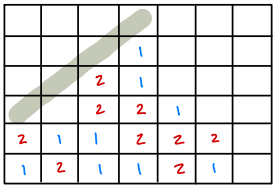
possible down right windows :

[0000]

windows meeting conditions : none so score = -62

step 5: calculating diagonal up right windows

diagonal $_0$:

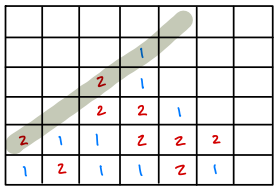


possible up right windows :

[0000]

windows meeting conditions : none so score = -62

diagonal $_1$:

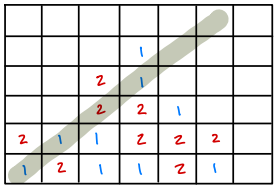


possible up right windows:

[2021], [0210]

windows meeting conditions: none so score = -62

diagonal $_2$:

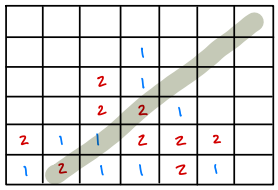


possible up right windows:

[1121], [1210], [2100]

windows meeting conditions: none so score = -62

diagonal $_3$:

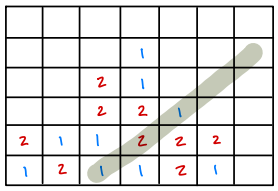


possible up right windows:

[2120], [1200], [2000]

windows meeting conditions: none so score = -62

diagonal $_4$:

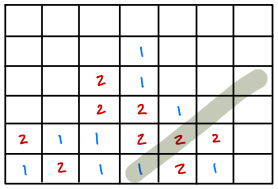


possible up right windows:

[1210], [2100]

windows meeting conditions: none so score = -62

diagonal $_5$:



possible up right windows:

[1200]

windows meeting conditions: none so score = -62

step 6: calculating final score

score = -62

Motivation: it rewards my AI for creating or extending sequences that might lead to a four in a row (winning) while at the same time "punishing" the opponent's efforts to do the same thing:

1. Center Bonus: The center column is generally more valuable because pieces placed there can participate in more potential winning combinations (horizontally, vertically, and diagonally) so we add points.
2. Immediate threats (windows of 4): also looking directly at groups of 4 cells (windows) across all possible directions.
    - If a window has multiple pieces from my AI and no opponent pieces, that window indicates a potential threat for my AI to win soon, so we add points.
    - If a window is mostly occupied by the opponent, we subtract points because it could mean a possible win for my AI.
3. Encourages strategic moving to maximize future opportunities. (Choosing the center column or blocking an opponent.
4. Immediate threats (windows of 5): In the case that my AI has three consecutive pieces and a 0 on either side, my AI proposes an immediate threat to my opponent. This is essentially because I have two opportunities at winning so I add points.

Combining these two ideas help my AI quickly assess how favorable a given board configuration is before making its next move.

Part III

```
Wins: 20 | Ties: 0 | Losses: 0 | Points: 20.0/20
```