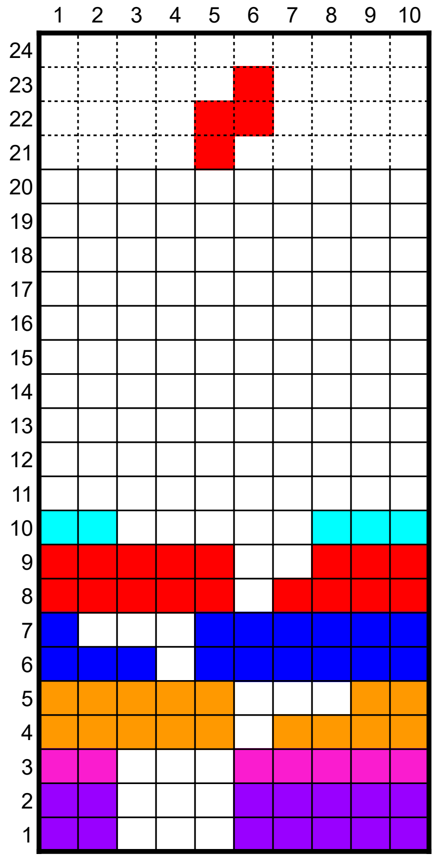
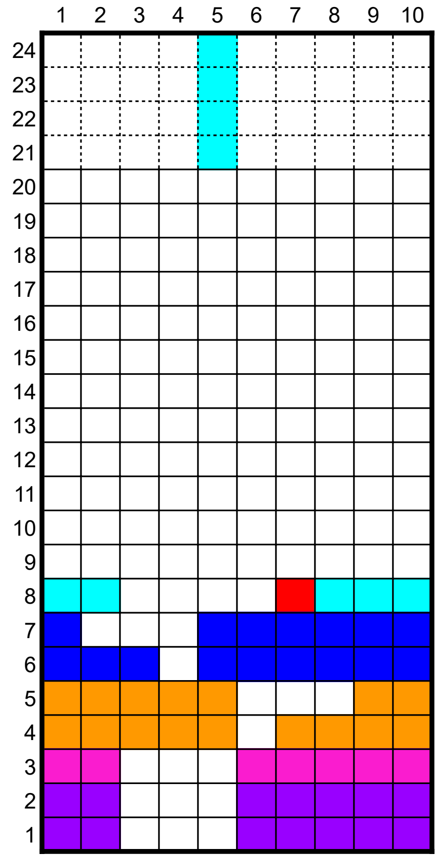
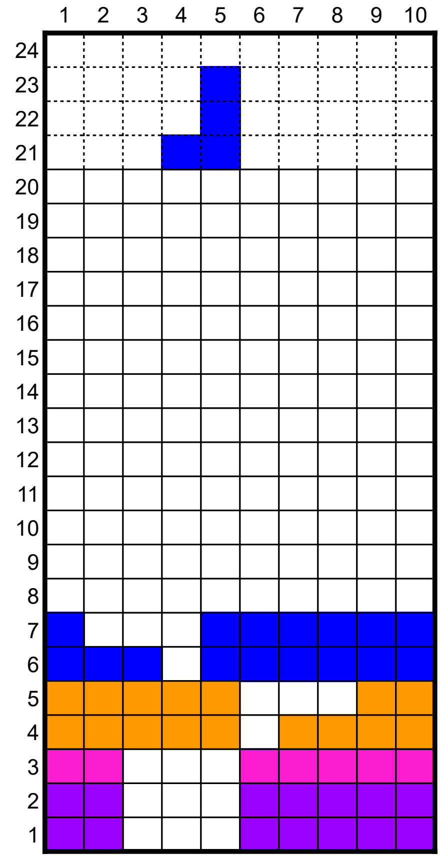
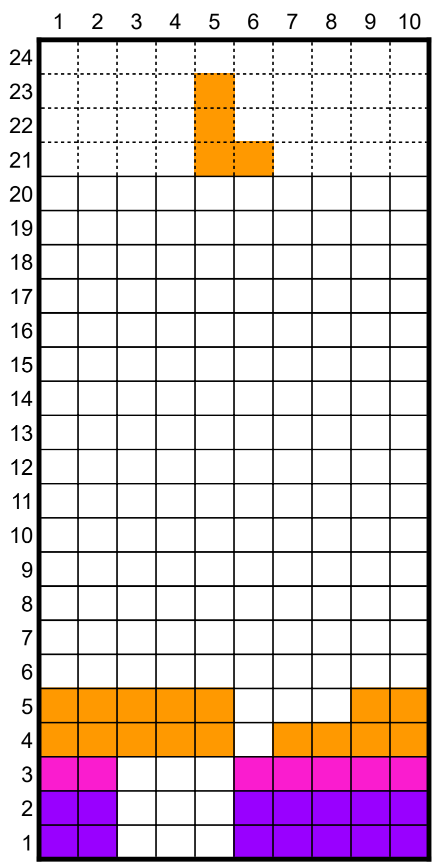
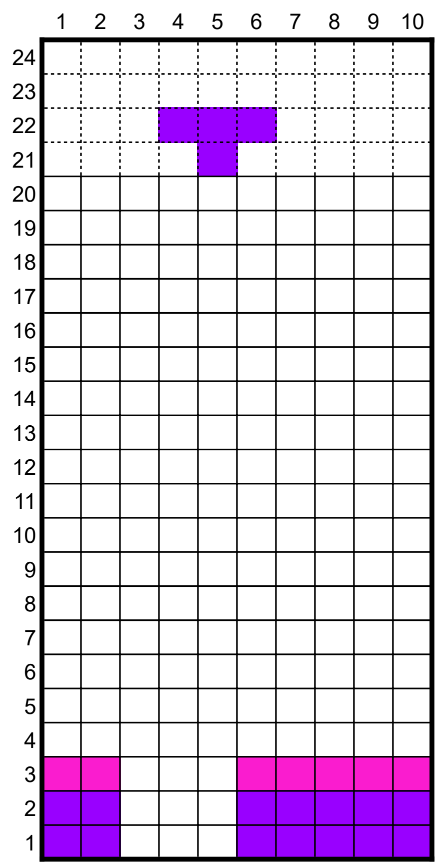
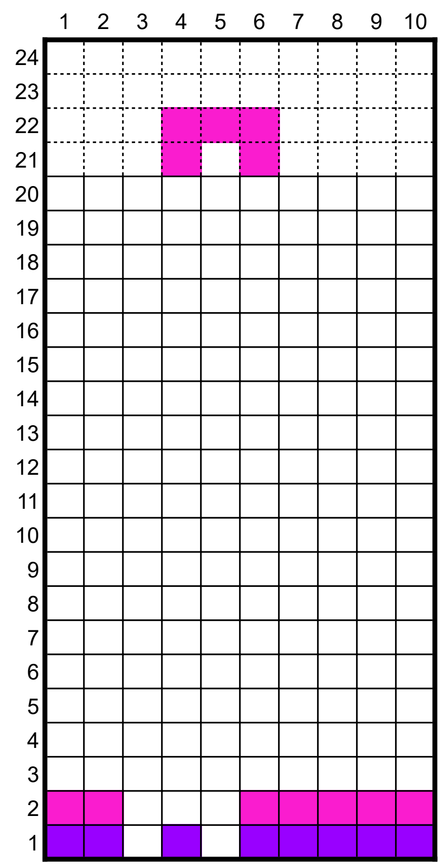

1 2 3 4 5 6 7 8 9 10
24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1


1 2 3 4 5 6 7 8 9 10
24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

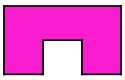

1 2 3 4 5 6 7 8 9 10
24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

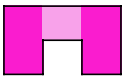

1 2 3 4 5 6 7 8 9 10
24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1


1 2 3 4 5 6 7 8 9 10
24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1


1 2 3 4 5 6 7 8 9 10
24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

ISAAC EBERT

Custom Piece "N": 

Rotation Point: 

1. My piece fits very closely to the original 7 Tetris pieces. It rotates around the middle point which makes it easy for players to adapt to. The piece will likely be used with "T" and "S" or "Z" pieces because of the one square that juts out when placed. The "N" piece allows for effective use with those pieces, allowing mutiple lines to be cleared with squares that may not be in ideal places. The piece also provides a challenge for the player if it gets placed sideways, as the player will have to figure out a way to get rid of it, possibly with a spin of the aforementioned tetrominoes.

2. The piece I made will be the "finisher" for the board. It cleans up the final two rows that were left by the prior piece. The end is a good example for the player to see the new piece's interaction with no so ideal board states.

3. The first and last piece placements need the player to think for a moment before they quick-drop the piece. If they place the "S" piece the wrong way or in the wrong spot, then the next piece will not clear the line like it is supposed to. Same goes with the "T" before the new piece, though that one will work in both the rotations that fit width-wise.

Although I did not end up using this custom board state, it still shows off the potential of what my piece can do.

My custom mechanic is a bounty system for points. On the screen there is an incentive for the player to complete a certain amount of lines to get bonus points. Each time a piece is placed, the bounty randomizes to another amount of lines. If the player completes the bounty, the score is multiplied by the amount shown on the cheatsheet.

Challenges faced while implementing:
- Grid coordinates for the piece cells. It took me a couple tries to get the piece looking right and the correct rotation. I used chart paper (excel) to finally figure out what I was doing wrong with the grid.
- Using events to trigger things in other scripts. I'm still pretty new with unity events as all I've used them for previously was scene management. I think what I have done works for this project, but I'm pretty sure I overcomplicated the scripts "talking" to each other for my bounty system.
- Getting my particle to appear where the last piece was placed when a row is cleared. I ended up putting the particle on the bounty text to show when a player completes the bounty without having to watch the score.

Link to video: https://youtu.be/cnXZfBV8rwg

Disclaimer: I'm not very good at Tetris, but I tried to show off the new mechanic and piece as best as I could.

Bounty Cheat-sheet

Line(s)	Without Bounty	With Bounty
1	**100**	**x10 = 1000**
2	**300**	**x8 = 2400**
3	**500**	**x6 = 3000**
4	**800**	**x4 = 3200**

ISAAC EBERT