Goal: Create a game to connect nodes from one corner of the board to the other by rotating them. This game will use a disconnected graph to connect these verticies. Win condition is cleared when a graph search reaches the opposing corner

**Game logic requirements:**

Create a list of Node objects referencable by index. These nodes should have variables for:

Direction, on a scale of either 0-3 or 1-4, which is read into the

Next location from a given node

Whether they are connected to the first node or not. This can be set with a smple Boolean

Methods to return and set these values.

Initial direction is randomized using randint() for each node.

Graph needs:

A dictionary for which node connects to which. This was done with a dictionary in class.

Rotating a node will delete the current connection from that node and point it to the new node on the node.

Graph search to tell if the end goal can be reached from the first node. If it can, then the game is won.

Points of note:

Nodes on the top and bottom rows and left and right columns will be able to point outside the board. These nodes should have their next target as None.

Using Pygame:

Most likely 50x50 pixels or a similar side and 100 nodes, 10x10

* Create a window and game loop for the game (done)
* Use rectangle collision to tell where the mouse has clicked (working progress). This function will return the index of the
* The image rotation is set to -90\*the rotation value (0-3) for the corresponding node.
* Need to add a method that switches which node image is used for connected and not connected nodes.

Other considerations for Pygame:

Keeping track of a turn counter possibly. This is an easy function that can simply incremement/decrement a value every time a

Only rotate nodes currently connected to the chain.

Win/Lose prompt, clicking on which starts a new game.

Closing window when the x is hit is already done.

In the event that we need more, we could make it two player and have it track if node 99 connects to node 0, alternating turns, but this is extremely low priority