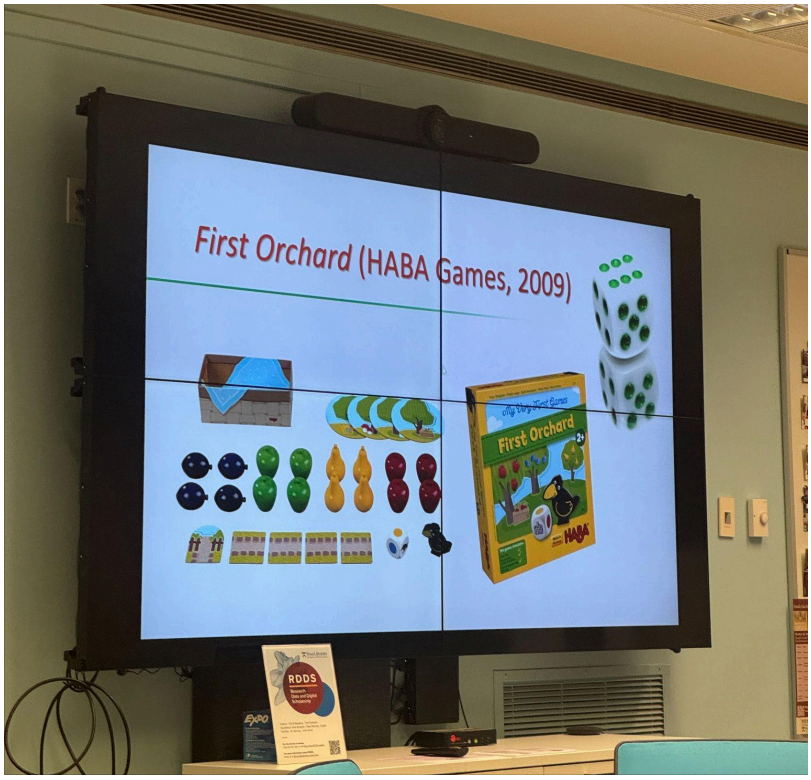
ENGL 1650: Penn Workshop – Computational Thinking & Board Games

Attending the Computational Thinking & Board Games Workshop at Penn was a fascinating experience, as it bridged the gap between board games and computer science principles. I had never thought of how a board game could relate to computational thinking, however after attending the workshop it is very clear how they align. The instructor of the workshop strategically emphasized the alignment of board games with computational principles, guiding us to see the connections practically. He began by asking us to name some of our favorite board games, creating an engaging atmosphere. We then jumped into how all of the board games mentioned have similar characteristics, on a broad level, that incorporate computational thinking. He went through each computer science principle, followed by a board game equivalent example.

The first principle was Random Number Generation, most commonly seen in board games through rolling a die, drawing a card, or spinning a spinner. This principle provides an element of randomness and chance, similar to a random computer-generated number. These random generated numbers affect outcomes in both computer science and board games; individuals need to think strategically, problem solve, and adapt depending on the number generated. Another computer science principle presented was Variables/Counting Variables, which is most commonly seen in board games through player scores., influence tracks, and hit points. Over time these variables change, reflecting a player's game status or progress, just like the updating and storing of certain data in coding. Conditionals are another computer science principle that are emphasized in board games from an if/then scenario. For example, if you roll 4+, the attack will hit, otherwise the attack misses. Booleans are another key computational principle that contains a binary outcome, like True/False or Yes/No; this relates to a board game in regards to its game states. For example, In Catan, a player's ability to build a new settlement can be represented through a True/False value, like whether or not the player has enough resources. Loops, another computer science principle, which deal with repeating something until a certain condition is met, are represented in board games in the most general form of a game loop. While all players' scores of the game are below 10, for example, they should continue playing. The last computer principle mentioned was the Program Code, which is represented by the rulebook in a board game. Just as the program code provides instructions for the computer to follow, the rulebook does the same for the players of a game. We finished off the workshop by playing a game that was designed for 2 year olds +, that despite its simplicity obtained many of the principles we discussed, including variables, loops, and random number generation.

This workshop definitely relates to this Introduction to Digital Humanities course, specifically the Tic-Tac-Toe exercise we took part in earlier this semester. In this exercise, we were asked to analyze Tic-Tac Toe and strategize new ways to play the game that may be just as simple or more complex than the original game. While we looked at ways to modify Tic-Tac-Toe, we touched on many concepts, like changing the game board, changing the amount of players involved, as well as introducing new symbols and rules, to name a few. This exercise ultimately emphasized the flexibility of game design and how minor adjustments can transform a game. The analysis of Tic-Tac-Toe, no doubt connects to broader computational principles, such as the

importance of rules, or program code, and the variety of outputs possible. Overall, the workshop was an engaging way to incorporate class material and understand how computational thinking can connect to real-world applications.




HABA
First Orchard
My Very First Games
2+