



(the actual book image itself got clipped, lol)

LPPBG: Welcome!

Learn Programming Programming Board Games uses peoples' existing knowledge about real world board games to explore topics in computer programming. The idea is to ground learning something new in something already known.

This project intends to be printed as a physical book.

Games in LPPB Games range from classic to contemporary.

LPPBG strives to be accessible, accurate, thorough, and interesting to programmers of all skill levels. Readers should especially be able to crack open the book to any board game they like and enjoy what they're reading without touching a computer.

The world of board games provides many tangible examples of topics directly connected to computer science and programming.

Example Insights:

In **Settlers of Catan**:

- Beginners will learn about how to calculate longest road with graph theory
- Experienced programmers may not have been exposed how to design a useful hex grid system or thought about how to implement trading in Catan.

In **Tic-Tac-Toe** beginner and experienced programmers alike may be too quick to try to over-optimize using for loops to check for three-in-a-row patterns.

Programmers of any skill level may be interested in reading examples about the balance between allowing players to freely manipulate a board vs locking players into actions without providing undo actions.

Programmers may just enjoy reading about how to program their favorite board game and agree or disagree about how the whole effort may be structured.

Programmers may want to read about how to solve stupid unprogrammable problems like how to answer "does your person look sketchy?" in a game of Guess Who.

And guess who would certainly benefit from finding out forcing players to manually place each of their ships at the start of a round of ASCII Battleship is a total waste of time.

The answer is you!!

Simple Games

These games are used to introduce fundamental programming principles like variables, data types, if statements, loops, and functions.

- Guess My Number
- Rock, Paper, Scissors
- Hangman

There must be more simple games, right?

Card Games

These games hammer in the idea of arrays and introduce the usefulness of classes.

- Mechanics
 - Representing a decks of cards with classes
 - Shuffling (Fischer-Yates)
- War
- Blackjack
- Poker (Single-player hand detection)
- Hearts
- Durak

Grid Games

Grid games are too common to not have their own section. Plus they are the perfect way to introduce 2D arrays. They work together well to show how variations on a theme can reuse some ideas from one program while introducing new ideas in another.

- Tic-Tac-Toe (2D arrays, redundancy, helper functions)
- Connect Four
- Checkers
- Chess
- Othello
- Go (data structures and algorithms)
- Chinese Checkers (uncommon geometries)

Fancy "Board" Games

These games have highly fanciful looking boards that they're actually played on. Let's talk about how to program something like that.

- Candyland
- Chutes and Ladders
- The Game of Life

Modern Games

Modern games show how knowledge of solved problems in data structures and computer science help guide our way to solving practical problems of our own.

- 1995 Settlers of Catan (hexes are hard, longest road is exactly graph theory)
- 2000 Carcassonne (graph theory helps again)
- 2004 Ticket to Ride (you got it, graph theory)
- 2008 Pandemic (cough, cough, I'm sick of graph theory)
- 2008 Dominion (How do you program playing cards that change the game?)

Racing Games

Just an obscure niche of board gaming that I personally like so of course I'll explore it.

- Formula D (actually this should be in the graphical board game section)
- Powerboats (expanding dealing with hexes, converting from single player to multiplayer)

That's all for now