A Lisp Interpreter

No, not that kind of lisp

What will I be creating?

An interpreter for a Lisp dialect

- Use existing guides to define…
    - Design and structure of the dialect itself
    - General stages for constructing the interpreter
- Independently…
    - Create the specific implementation
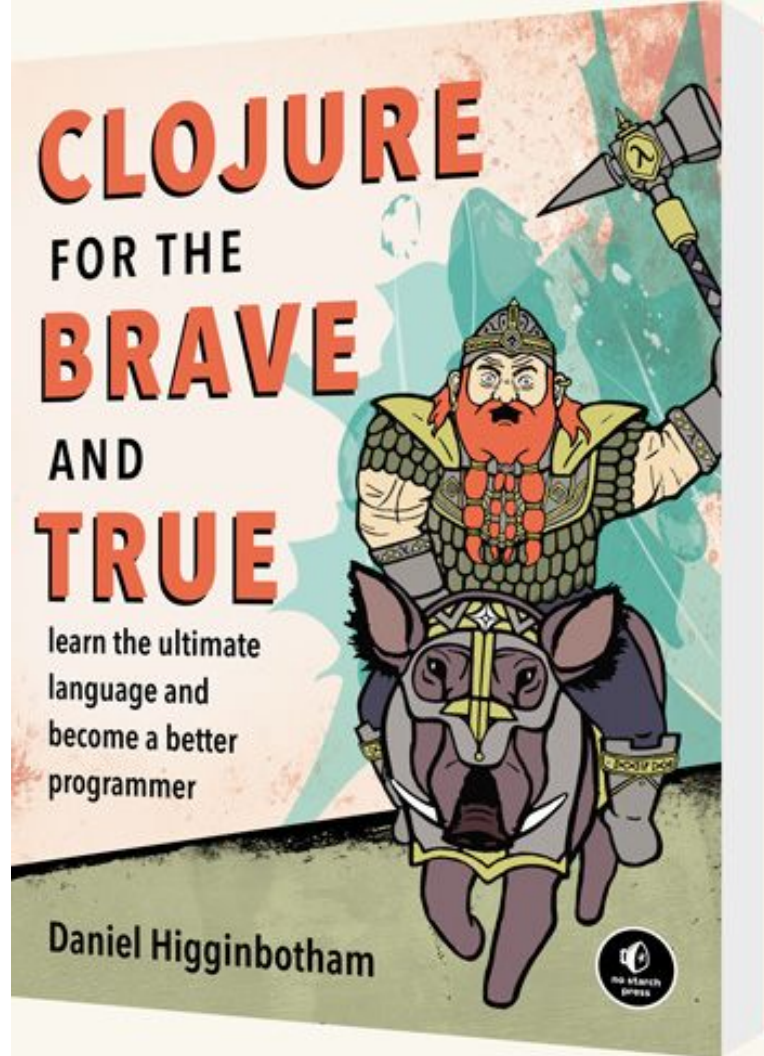   - Create additions as seen fit

What will I be reading?

Clojure for the Brave and True

- Learn Clojure, a Lisp dialect
- Gain insight into the Lisp language family
- Use Clojure to create my project

Crafting Interpreters

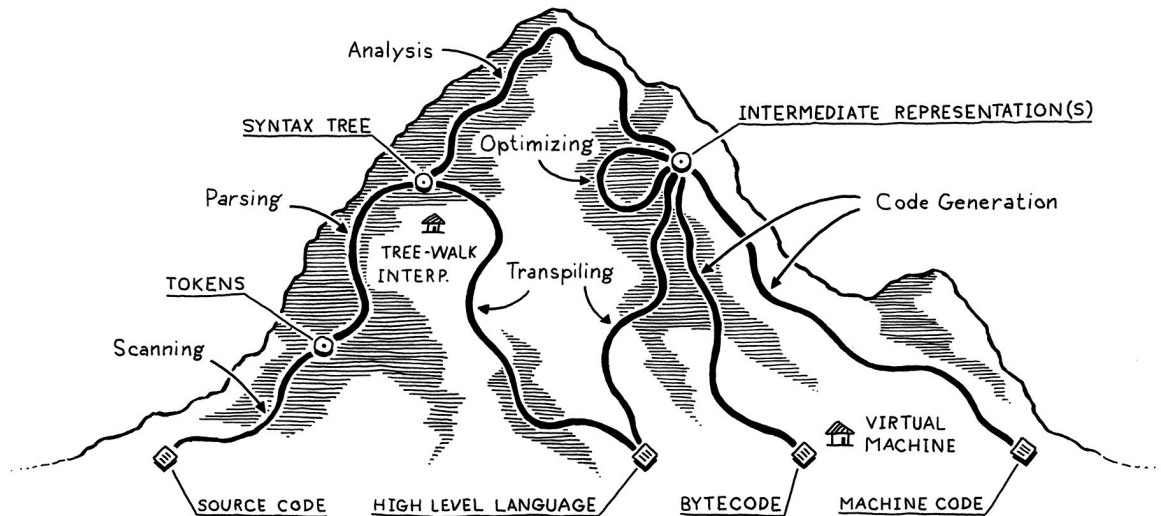- Learn theory and implementation techniques behind interpreters

What will I be writing?

- Notes for both books
- Documentation of my process
  - How I tackled each step
- Progress log
  - Revisions made each day
  - Major steps and milestones
- Reflection on the entire process
  - Pros and cons of the experience
  - Suggestions for improvement
- Lots and lots of code!

What will I be researching?

- Lisp family of languages
  - Origins and foundations
  - General design choices
  - Iconic features
- Language interpreters
  - Relevant linguistics concepts
  - Stages of interpretation

Analysis

SYNTAX TREE

INTERMEDIATE REPRESENTATION(S)

Optimizing

Parsing

TREE-WALK
INTERP.

Code Generation

TOKENS

Transpiling

Scanning

VIRTUAL
MACHINE

SOURCE CODE

HIGH LEVEL LANGUAGE

BYTECODE

MACHINE CODE

What will I be presenting?

- Explanation of the language and its features
- Demonstration of interpreter
  - At minimum…
    - Basic math
    - User interaction
- Explanation of the stages of interpretation
  - Relevant linguistics
    - Parallels between natural and programming languages
  - Entire process of code ➜ execution

Why is this a worthy project?

- Vested interest in programming languages
    - Design and implementation
    - Desire to pursue in college and career
- Conveying complex ideas to unfamiliar audience
    - Integral to a successful CS career
    - Reduce the mystery surrounding programming languages