

Part 3: *AST* interpretation

Interpretation overview

- As discussed, we will be using a tree-walk interpreter
- This means that we will start at the top-level AST node returned by the parser, and recursively process each of its children
- This could be done with a single function that conditions on the type of the AST node, but is better served by having a function for each type of AST node instead
- In addition, we will have a general "process AST node" function that calls the relevant function for that type
- Note: it may be better to do this with class methods, but we haven't gone too in-depth on how classes work, so we will only be using functions