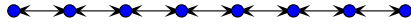


Notes on Programming Languages



Basic Scheme

- *basics*
- *evaluation of Scheme expressions*

Scheme and Functions

- *recursive vs. iterative processes*
- *bindings*
- *functions*
- *combining functions*

Scheme and Data

- *data structures in Scheme*
- *data abstraction*
- *simple list manipulations*

Abstraction with Data

- *conventional interfaces*
- *data-directed programming using tagged data*
- *encapsulation*

State

- *state*
- *drawbacks to state*

User-defined Data Structures with State

- *stacks*
- *queues*

Interesting Applications

- *discrete-event simulation*
- *constraint-based programming* (optional)
- *introduction to streams*
- *more on streams*
- *still more on streams*

Topics concerning the implementation of a programming language

- *overview*
- *grammars*
- *fun with grammars*
- *lexical analysis (scanning)*
- *recursive descent parsing for language recognition*
- *parse trees and pretty printing*

- *implementing environments*
- *evaluating parse trees*
- *adding built-in functions*
- *associativity and precedence*
- *detecting undefined variables*
- *static typing versus dynamic typing*
- *object orientation*

Other Programming Language Topics

- *invariants*
- *axiomatic semantics*
- *more on axiomatic semantics*
- *still more on axiomatic semantics*
- *variations on passing arguments*