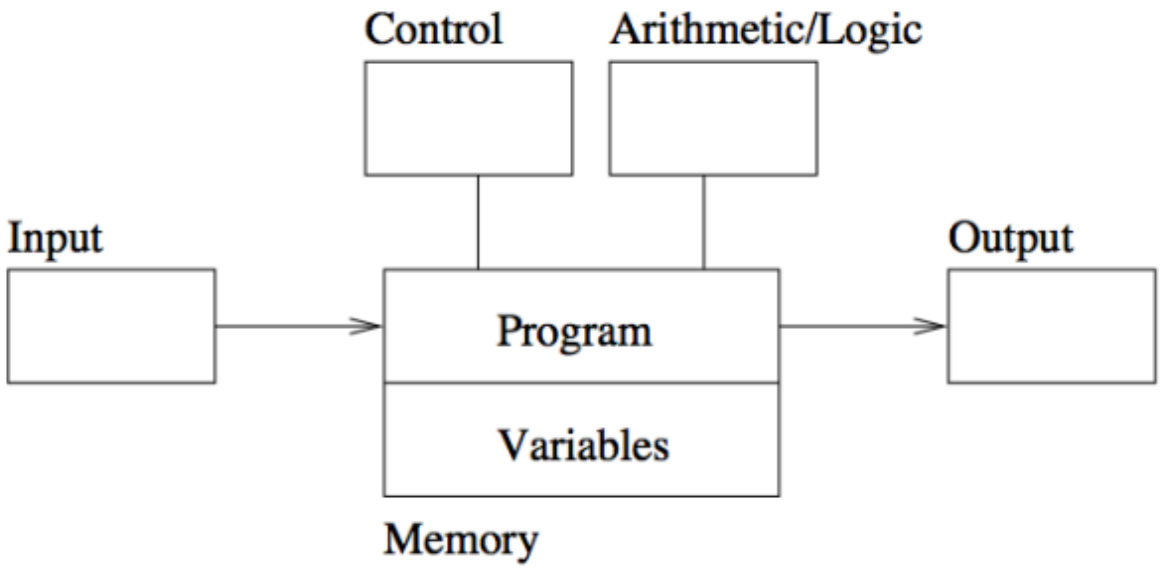
Programming Paradigms

The Von Neumann-Eckert Model



Paradigms

- Modern languages are multi-paradigmatic
  - Examples
    - Haskell (F + I)
    - Scala (F + I + O)
    - OCaml (F + I + O)
    - F Sharp (F + I + O)
    - Python (F + I + O)

Imperative Paradigm

- Program and data are indistinguishable in memory
- Program
  - Sequence of commands
- State
  - Values of all variables when program runs
- Large programs use procedural abstraction
- Examples
  - Cobol, Fortran, C, Ada, Perl

Object Oriented Paradigm

- An OO program is a collection of objects that interact by passing messages that transform state
- Sending messages
- Inheritance
- Polymorphism
- Examples
    - Smalltalk, Java, C++, C#, Python

Functional Paradigm

- Models a computation as a collection of mathematical functions
    - Input = domain
    - Output = range
- Characterized by
    - Functional composition
    - Recursion
    - No state changes
    - No variable assignments
    - Mathematical
        - Output results instantly
- Examples
    - Lisp, Scheme, ML, Haskell

Logic Paradigm

- Declares what outcome should be accomplished, rather than how it should be accomplished
- Characterized by
    - Programs as sets of constraints on a problem
    - Programs that achieve all possible solutions
    - Programs that are nondeterminate