Formal Grammars

• Backus-Naur Form (BNF)
• Express syntax using patterns
• Expression and value domains
Halting Problem

- Incompleteness:
  - Any sufficiently complex language can express truths it cannot prove
  - Non-computable functions

- Many interesting program properties are non-computable
  - “Does P(x) halt?”
  - “Does P(x) read user input?”
Continuations

• call/cc
• Time machine
• Browser back button

• Lecture recap 2:30 pm today, TCL 308
Functional Programming

- Easier to understand and analyze
- Enables lazy evaluation
- Enables high-performance stream computing
Stream Programming

Arithmetic Logic Units

ALU

7 8 2 4 5

Memory

8 9 3 5 6
(define (wrap scheme-proc)
  (BUILTIN-PROC
   (λ (eph-values)
      (->VAL
       (apply scheme-proc
         (map ->Scheme eph-values))))))
• Abstract computation
• Delay computation
• Capture variables
Y-Combinator

\[ Y = (\lambda (g) \left( (\lambda (z) (z \ z)) \left( (\lambda (x) (g \ (\lambda (y) ((x \ x) \ y)))) \right) \right) \right) \]

factorial = \( Y \ (\lambda (f) \ (\lambda (n) \left( (\text{if} \ (\text{zero?} \ n) \ 1 \ (* \ n \ (f \ (\text{sub1} \ n)))))))) \)
MACROS

- Parser plugins
- Extend syntax of language
- e.g., \( (OR \ a \ b) \Rightarrow ? \)
Learning a Language

- Delay computation
- Capture variables
- Apply delayed computation
- Mutate variables
- Control flow
- Extend syntax