

# OCaml!

CS 242

October 9, 2017

# Feedback

- **Assignments**
  - Handout not well documented
  - Systems concepts unfamiliar to students
  - Taking too long relative to course units
  - More thorough tests
- **Lectures**
  - Generally positive reaction!
  - Access to materials used in class
  - Keeping questions on track
  - Language practice vs. language theory

# Unit goals

- **Core tenets of functional programming**
  - Immutability, functions everywhere, rich type system
  - Putting OCaml into practice
- **Basic theory of programming languages**
  - Formalized type systems and runtime semantics
  - Focus on proving language safety
- **Advanced abstractions for data and control**
  - Data: algebraic data types, polymorphic/existential types
  - Control: exceptions, continuations, coroutines

# Today's goals

- **OCaml syntax and semantics**
- **Functional programming model**

# OCaml is a functional PL

- **Functional PLs are widely studied in academia**
  - Haskell, OCaml, Standard ML are all over 20 years old!
  - Embody many principles of academic PLs
- **Yet to see widespread adoption**
  - But ideas trickle slowly into the mainstream

**Let's learn  
OCaml!**