CS 476: Programming Language Design
Questions?

Top
Summary

• We’ve looked at PLs as designers (what’s in a language? how should it work?) and implementers (how do we get a computer to run it?)

• We described PLs with math (grammars, type rules, big- and small-step semantics rules) and turned those rules into OCaml code (typecheckers, interpreters, etc.)

• We examined the features of:
  — Imperative languages (variables, control flow, function calls)
  — Object-oriented languages (objects, inheritance, references)
  — Functional languages (functions as values, pattern-matching, type inference)
  — And a few others
Summary

• Now you can:
  • Write OCaml code (or code in another functional language)
  • Describe PL features
  • Translate inference rules into code
  • Figure out what happens when you combine features from different languages
  • Implement a language of your own, or add a new feature to an existing language design
Questions?

Top