Lambda Calculus

Paul Gustafson

Math 482 - Texas A&M University

March 5, 2013

Why is the λ-calculus important?

- Computer Science - Variable binding, scope
- Typed λ-calculi are the basis for modern type systems in programming
- Functional programming languages (Algol, Lisp, ML, Haskell)
- Recursion theory/computability
- Linguistics

How was the λ-calculus developed?

- Formal system of logic developed by Alonzo Church in 1932
- Used to address the *Entscheidungsproblem* – “Is every predicate first-order logic provable?”
- Similar developments - Godel numbers, Turing machines

Conversion Rules

- α-conversion: $\lambda x.[...x...] = \lambda y.[...y...]$.
  "We can rename dummy variables."
- Example: $\lambda a.a = \lambda b.b$
- β-conversion: $\lambda x.[...x...]T = [...T...]$.
  "Evaluation by substitution."
- Example: $(\lambda x.x)y = y$.
- η-conversion: $\lambda x.F(x) = F$.
  "Extensionality - a function is defined by what it does."
  Controversial; often left out of compilers.
- Example: $\lambda y.\lambda x.yx = \lambda y.y$

Church numerals

- $0 := \lambda f.\lambda x.x$
- $1 := \lambda f.\lambda x.fx$
- $2 := \lambda f.\lambda x.f(fx)$
- $3 := \lambda f.\lambda x.f(f(fx))$
- $\ldots$