CS 773 Winter 2001

07. Higher-Order Logic

Instructor: W. M. Farmer

Revised: 20 March 2001

What is Higher-Order Logic?

- Higher-order functions (or predicates) can be represented by terms
  - Note: A function $f : A \to B$ is **higher-order** if A or B contains functions
- Quantified variables can range over functions
- Types or sorts are used to:
  - Organize the functions of the logic
  - Control the formation of expressions
  - Classify expressions by value

Type Theory

- A higher-order logic can be viewed as a theory of types
- Russell introduced a logic called the **Theory of Types (TT)** in 1908 to serve as a foundation for mathematics
  - Included a hierarchy of types to avoid set-theoretic paradoxes like Russell's Paradox
  - Employed as the logic of Whitehead and Russell's *Principia Mathematica*
  - Not used today due to its high complexity
- Chwistek and Ramsey suggested a simplified version of TT called **Simple Type Theory (STT)** in the 1920s
  - A formulation of STT with lambda-notation was introduced by Church in 1940

Intuitionistic Type Theory

- Several intuitionistic or constructive type theories have been developed
- Examples:
  - Martin-Löf's **Intuitionistic Type Theory** (1980)
  - Coquand and Huet's **Calculus of Constructions** (1984)
- Many intuitionistic type theories exploit the Curry-Howard Formulas-as-Types Isomorphism
  - Formulas serve as types or specifications
  - Terms serve as proofs or programs

Syntax of STT

- The set $\mathcal{T}$ of **types** of STT is defined inductively by:
  - ι (type of individuals) is a type
  - $*$ (type of truth values) is a type
  - If α and β are types, then $(\alpha \to \beta)$ is a type
- A **language** of STT is a tuple $L = (\mathcal{V}, \mathcal{C}, \tau)$ where:
  - $\mathcal{V}$ is an infinite set of **variables**
  - $\mathcal{C}$ is a set of **constants**
  - $\tau : \mathcal{C} \to \mathcal{T}$ is a total function
- The set of **expressions** of L is defined inductively

Semantics of STT

- Three kinds of semantics:
  - Standard: Terms are defined, formulas are two-valued
  - Partial: Terms may be undefined, formulas are two-valued
  - Three-valued: Formulas are three-valued
- A **model** of a language $L = (\mathcal{V}, \mathcal{C}, \tau)$ of STT is a pair $M = (\mathcal{D}, I)$ where:
  - $\mathcal{D} = \{D_\alpha : \alpha \in \mathcal{T}\}$
  - D_ι is nonempty and $D_* = \{\mathrm{T}, \mathrm{F}\}$
  - $D_{(\alpha \to \beta)}$ is the set of functions from D_α to D_β
  - I maps each $c \in \mathcal{C}$ to an element of $D_{\tau(\alpha)}$
- The **valuation** function for L in M is defined inductively

Extensions to STT

- Types
  - Additional type constants and constructors
  - Type variables
  - Subtypes
- Expressions
  - Additional expression constants and constructors
  - Multivariate functions