

# Lambda Calculus

## Abstract Syntax

$X = \{x, y, z, \dots\}$  — variables

$C = \{X, Y, Z, \dots\}$  — constructors

$\Lambda =$	$X$	— variable
	$  \quad C$	— constructor
	$  \quad \lambda X. \Lambda$	— abstraction
	$  \quad \Lambda \Lambda$	— application

## Simple Typing á la Curry

Types:

$\mathcal{T} = A \mid \mathcal{T} \rightarrow \mathcal{T}$

where  $A$  is a set of primitive types.

Environment  $\Gamma : X \rightarrow \mathcal{T}$  is an assignment of types to (some) variables.

Typing judgements:

$\Gamma \vdash x : \Gamma x$  [VAR]

$$\frac{\Gamma, x : \tau \vdash M : \rho}{\Gamma \vdash \lambda x. M : \tau \rightarrow \rho}$$
 [ABSTR]

$$\frac{\Gamma \vdash M : \tau \rightarrow \rho, \Gamma \vdash N : \tau}{\Gamma \vdash M N : \rho}$$
 [APP]