Proposições como Tipos

Marcos Benevides

Universidade Federal do Maranhão

marcos.schonfinkel@gmail.com
https://github.com/mschonfinkel/PropositionsAsTypes

GELF - Grupo de Estudos em Lógica e Filosofia Formal

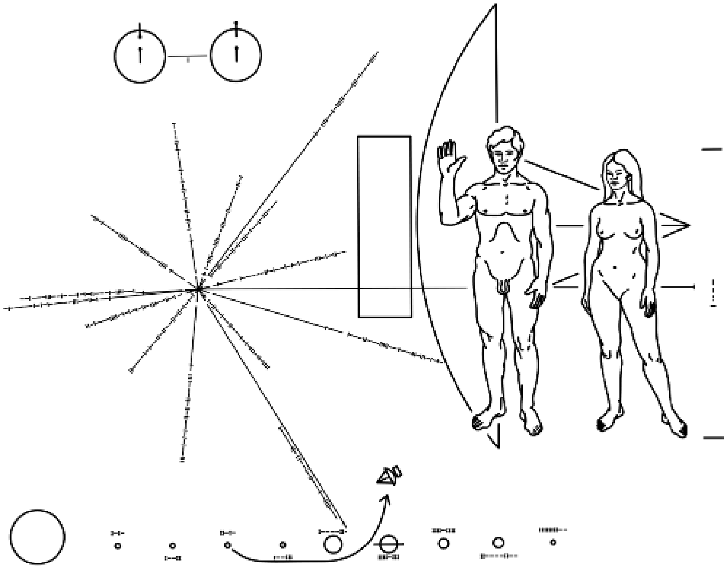
Baseado no artigo original [Wadler, 2015] e apresentação de Philip Wadler.

13 de Julho de 2018

Overview I

> "Propositions as Types is a notion with depth. It describes a correspondence between a given logic and a given programming language."
>
> — Propositions as Types, Wadler

A Placa Pioneer



Algoritmos

(a) Euclides (325–265 BCE)

(b) Al-Khwarizmi (780–850 CE)

Computabilidade

Apesar de ser uma idéia antiga, definições formais só foram aparecer a partir do Séc. XX.

- Alonzo Church: λ-Cálculo (1935)
  [Church, 1936]
- Kurt Gödel: Funções Recursivas (1935)
  Propostas por Gödel em Princeton e publicadas por Kleene em [Kleene, 1936]
- Alan Turing: Máquinas de Turing (1936)
  [Turing, 1936]

Por que isso aconteceu?

(c) David Hilbert

(d) Entscheidungsproblem

Por que isso aconteceu?

Um dos objetivos do Programa de Hilbert era resolver o Entscheidungsproblem (problema de decisão), ou seja, desenvolver um procedimento "efetivamente calculável"para determinar a verdade ou a falsidade de qualquer proposição.

Por que isso aconteceu?

Um dos objetivos do Programa de Hilbert era resolver o Entscheidungsproblem (problema de decisão), ou seja, desenvolver um procedimento "efetivamente calculável"para determinar a verdade ou a falsidade de qualquer proposição.

Mas para resolver o Entscheidungsproblem é necessário uma definição formal de "efetivamente cálculável".

Gerhard Gentzen, 1909 - 1945.

First I wanted to construct a formalism which comes as close as possible to actual reasoning. Thus arose a "calculus of natural deduction".

Dedução Natural

Introdução

A notação

$$\begin{array}{c} \vdots \\ A \end{array}$$

denota a dedução de A, terminando em A.

Regras

- *Hypothesis*: A
- *Introductions*:

$$\dfrac{\begin{matrix}\vdots & & \vdots \\ A & & B\end{matrix}}{A \wedge B}\,\wedge\mathcal{I} \qquad \dfrac{\begin{matrix}[A] \\ \vdots \\ B\end{matrix}}{A \Rightarrow B}\,\Rightarrow\mathcal{I} \qquad \dfrac{\begin{matrix}\vdots \\ A\end{matrix}}{\forall\xi.\,A}\,\forall\mathcal{I}$$

- *Eliminations*:

$$\dfrac{\begin{matrix}\vdots \\ A \wedge B\end{matrix}}{A}\,\wedge 1\mathcal{E} \qquad \dfrac{\begin{matrix}\vdots \\ A \wedge B\end{matrix}}{B}\,\wedge 2\mathcal{E} \qquad \dfrac{\begin{matrix}\vdots & & \vdots \\ A & & A \Rightarrow B\end{matrix}}{B}\,\Rightarrow\mathcal{E} \qquad \dfrac{\begin{matrix}\vdots \\ \forall\xi.\,A\end{matrix}}{A[a/\xi]}\,\forall\mathcal{E}$$

Figura: Fonte: [Girard, 1989]

Exemplo

Proposição: $(B \wedge A) \Rightarrow (A \wedge B)$
Prova:

$$\dfrac{\dfrac{\dfrac{[B \wedge A]}{A}\wedge 2\mathcal{E} \qquad \dfrac{[B \wedge A]}{B}\wedge 1\mathcal{E}}{A \wedge B}\wedge\mathcal{I}}{(B \wedge A) \Rightarrow (A \wedge B)}\Rightarrow\mathcal{I}$$

Simplificando Provas

$$\dfrac{\dfrac{\begin{matrix}[A]\\ \vdots \\ B\end{matrix}}{A \Rightarrow B}\Rightarrow\mathcal{I} \qquad \begin{matrix}\vdots\\ A\end{matrix}}{B}\Rightarrow\mathcal{E}$$

$$\begin{matrix}\vdots\\ \mathrm{A}\\ \vdots\\ \mathrm{B}\end{matrix}$$

Simplificando Provas

$$\dfrac{\dfrac{\begin{matrix}\vdots & & \vdots \\ A & & B\end{matrix}}{A \wedge B}\wedge\mathcal{I}}{A}\wedge 1\mathcal{E} \qquad\qquad \begin{matrix}\vdots \\ A\end{matrix}$$

Simplificando Provas

$$\dfrac{\dfrac{\dfrac{\dfrac{[B \wedge A]}{A}\,\wedge 2\mathcal{E} \qquad \dfrac{[B \wedge A]}{B}\,\wedge 1\mathcal{E}}{A \wedge B}\,\wedge\mathcal{I}}{(B \wedge A) \Rightarrow (A \wedge B)}\,\Rightarrow\mathcal{I} \qquad \dfrac{[B] \qquad [A]}{B \wedge A}\,\wedge\mathcal{I}}{A \wedge B}\,\Rightarrow\mathcal{E}$$

Simplificando Provas

$$
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{[B \wedge A]}{A}\,\wedge 2\mathcal{E}
      \qquad
      \dfrac{[B \wedge A]}{B}\,\wedge 1\mathcal{E}
    }{A \wedge B}\,\wedge\mathcal{I}
  }{(B \wedge A) \Rightarrow (A \wedge B)}\,\Rightarrow\mathcal{I}
  \qquad
  \dfrac{[B] \qquad [A]}{B \wedge A}\,\wedge\mathcal{I}
}{A \wedge B}\,\Rightarrow\mathcal{E}
$$

Simplificando Provas

$$\cfrac{\cfrac{\cfrac{\cfrac{[B \wedge A]}{A}\,\wedge 2\mathcal{E} \qquad \cfrac{[B \wedge A]}{B}\,\wedge 1\mathcal{E}}{A \wedge B}\,\wedge\mathcal{I}}{(B \wedge A) \Rightarrow (A \wedge B)}\,\Rightarrow\mathcal{I} \qquad \cfrac{[B] \qquad [A]}{B \wedge A}\,\wedge\mathcal{I}}{A \wedge B}\,\Rightarrow\mathcal{E}$$

$$\begin{array}{c}\cfrac{[B] \qquad [A]}{B \wedge A} \\ \vdots \\ A \wedge B\end{array}$$

Simplificando Provas

$$
\dfrac{\dfrac{\dfrac{[B] \qquad [A]}{B \wedge A}\,{\color{red}\wedge\mathcal{I}}}{\color{red}A}\,{\color{red}\wedge 2\mathcal{E}} \qquad \dfrac{\dfrac{[B] \qquad [A]}{B \wedge A}\,{\color{blue}\wedge\mathcal{I}}}{\color{blue}B}\,{\color{blue}\wedge 1\mathcal{E}}}{A \wedge B}
$$

Simplificando Provas

$$\dfrac{\dfrac{\dfrac{[B] \qquad [A]}{B \wedge A}\wedge\mathcal{I}}{A}\wedge 2\mathcal{E} \qquad \dfrac{\dfrac{[B] \qquad [A]}{B \wedge A}\wedge\mathcal{I}}{B}\wedge 1\mathcal{E}}{A \wedge B}$$

$$\dfrac{[A] \qquad [B]}{A \wedge B}\wedge\mathcal{I}$$

Sequent Calculus

Baseado (roubado!) no tutorial interativo de [Yang, 2012].

λ-Cálculo

$$s ::= x \mid (\lambda x.s) \mid (s\ s)$$

onde x é uma variável atômica.

Combinador Y

$$Y = \lambda f.(\lambda x.(f(xx))\ \lambda x.(xx)) \tag{1}$$

Descoberto por Haskell Curry.

Paradoxos = Loops Infinitos

"Whereas self-application in Russell's logic leads to paradox, self-application in Church's untyped λ-calculus leads to non-terminating computations."

— Propositions as Types, Wadler

Alonzo Church, 1903 - 1995.
'There may, indeed, be other applications of the system than its use as a logic.'

λ-Cálculo Simplesmente Tipado

$$\dfrac{\begin{array}{c}[x : A] \\ \vdots \\ N : B\end{array}}{\lambda x.N : A \Rightarrow B} \Rightarrow \mathcal{I}$$

$$\dfrac{L : A \Rightarrow B \qquad M : A}{N : B} \Rightarrow \mathcal{E}$$

λ-Cálculo Simplesmente Tipado

$$\frac{M : A \qquad N : B}{(M, N) : A \wedge B}\wedge\mathcal{I} \qquad \frac{L : A \wedge B}{\texttt{fst}\ L : A}\wedge 1\mathcal{E} \qquad \frac{L : A \wedge B}{\texttt{snd}\ L : B}\wedge 2\mathcal{E}$$

λ-Cálculo Simplesmente Tipado

$$\dfrac{\begin{array}{c}[x : A] \\ \vdots \\ N : B\end{array}}{\lambda x.N : A \Rightarrow B} \Rightarrow \mathcal{I}$$

$$\dfrac{L : A \Rightarrow B \qquad M : A}{N : B} \Rightarrow \mathcal{E}$$

λ-Cálculo Simplesmente Tipado

$$\frac{M : A \qquad N : B}{(M, N) : A \wedge B} \wedge\mathcal{I} \qquad \frac{L : A \wedge B}{\texttt{fst}\ L : A} \wedge 1\mathcal{E} \qquad \frac{L : A \wedge B}{\texttt{snd}\ L : B} \wedge 2\mathcal{E}$$

Programa

$$\dfrac{\dfrac{\dfrac{z : B \wedge A}{\texttt{snd}\ \ z : A}\wedge 2\mathcal{E} \qquad \dfrac{z : B \wedge A}{\texttt{fst}\ \ z : B}\wedge 1\mathcal{E}}{(\texttt{snd}\ \ z, \texttt{fst}\ \ \texttt{z}) : (A \wedge B)}\wedge\mathcal{I}}{\lambda z.(\texttt{snd}\ \ \texttt{z}, \texttt{fst}\ \ \texttt{z}) : (B \wedge A) \Rightarrow (A \wedge B)}\Rightarrow\mathcal{I}$$

Valorando programas

$$\dfrac{\dfrac{\begin{array}{c}[x : A]\\ \vdots \\ N : B\end{array}}{\lambda x.N : A \Rightarrow B}\Rightarrow \mathcal{I} \qquad \begin{array}{c}\vdots \\ M : A\end{array}}{(\lambda x.N)\ M : B}\Rightarrow \mathcal{E}$$

$$\begin{array}{c}\vdots \\ M : A \\ \vdots \\ N\{M/x\} : B\end{array}$$

Valorando programas

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ M : A\end{array} \qquad \begin{array}{c}\vdots\\ N : B\end{array}}{(M, N) : A \wedge B}\,\wedge\mathcal{I}}{\texttt{fst (M,N):A}}\,\wedge 1\mathcal{E}$$

$$\begin{array}{c}\vdots\\ M : A\end{array}$$

Computação

$$\dfrac{\dfrac{\dfrac{\dfrac{[z : B \wedge A]}{\texttt{snd z:A}}\wedge 2\mathcal{E} \qquad \dfrac{[z : B \wedge A]}{\texttt{fst z:B}}\wedge 1\mathcal{E}}{(\texttt{snd z}, \texttt{fst z}) : A \wedge B}\wedge\mathcal{I}}{\lambda z.(\texttt{snd z}, \texttt{fst z}) : (B \wedge A) \to (A \wedge B)}\Rightarrow\mathcal{I} \qquad \dfrac{[y : B] \qquad [x : A]}{(y, x) : B \wedge A}\wedge\mathcal{I}}{(\lambda z.(\texttt{snd z}, \texttt{fst z}))(y, x) : (A, B)}\Rightarrow\mathcal{E}$$

Computação $\leftrightarrows$ Simplificação de Provas

$$
\dfrac{
  \dfrac{
    \dfrac{[y : B] \qquad [x : A]}{(y,x) : B \wedge A}\,\wedge\mathcal{I}
  }{\texttt{snd (y,x):A}}\,\wedge 2\mathcal{E}
  \qquad
  \dfrac{
    \dfrac{[y : B] \qquad [x : A]}{(y,x) : B \wedge A}\,\wedge\mathcal{I}
  }{\texttt{fst (y,x):B}}\,\wedge 1\mathcal{E}
}{(\texttt{snd z}, \texttt{fst z}) : A \wedge B}\,\wedge\mathcal{I}
$$

Computação $\leftrightarrows$ Simplificação de Provas

$$\frac{[x : A] \qquad [y : B]}{(x, y) : A \wedge B} \wedge\mathcal{I}$$

Alan Turing, 1912 - 1954.

An early proof of normalization.

O λ-Cálculo não tipado permite a definição de qualquer função efetivamente computável, mas possui um problema da parada insolúvel. Em contrapartida, o λ-Cálculo sem recursão geral (*general recursion*) possui um Problema da Parada trivial, pois todo programa termina, mas não consegue representar todas as funções computáveis.

Curry-Howard

"Algorithms are the computational content of proofs."

— Robert Harper

"Curry–Howard is a double-barreled name that ensures the existence of other double-barreled names."

— Philip Wadler

Conjunção

"$A \wedge B$ corresponds to Cartesian product $A \times B$, that is, a record with two fields, also known as a pair. A proof of the proposition $A \wedge B$ consists of a proof of A and a proof of B. Similarly, a value of type $A \times B$ consists of a value of type A and a value of type B."

Disjunção

"$A \vee B$ corresponds to a disjoint sum $A + B$, that is, a variant with two alternatives. A proof of the proposition $A \vee B$ consists of either a proof of A or a proof of B, including an indication of which of the two has been proved. Similarly, a value of type $A + B$ consists of either a value of type A or a value of type B, including an indication of whether this is a left or right summand."

Implicação

"$A \Rightarrow B$ corresponds to function space $A \to B$. A proof of the proposition $A \Rightarrow B$ consists of a procedure that given a proof of A yields a proof of B. Similarly, a value of type $A \to B$ consists of a function that when applied to a value of type A returns a value of type B."

$P \wedge (Q \vee R) \Rightarrow (P \wedge Q) \vee (P \wedge R)$

$$A \vee B \longmapsto A + B$$

$$A \wedge B \longmapsto A \times B$$

$$\frac{x : A \qquad y : B}{(x, y) : A \times B}$$

$$\frac{x : A}{\texttt{Left}\ \ x : (A + B)}$$

$$\frac{y : B}{\texttt{Right}\ \ y : (A + B)}$$

$$A \Rightarrow B \longmapsto A \rightarrow B$$

Construir uma função $f : A \rightarrow B$.

$P \times (Q + R) \to (P \times Q) + (P \times R)$

$$f(x,\ \texttt{Left}\ \ y) = \texttt{Left}\ \ (x, y)$$
$$f(x,\ \texttt{Right}\ \ z) = \texttt{Right}\ \ (x, z)$$

Isomorfismos

Isomorfismos

Dedução Natural (Gentzen, 1935)	$\longleftrightarrow$	λ-Cálculo Tipado (Church, 1940)

Isomorfismos

Dedução Natural (Gentzen, 1935)	$\longleftrightarrow$	λ-Cálculo Tipado (Church, 1940)
Type Schemes (Hindley, 1969)	$\longleftrightarrow$	Sistema de Tipos da ML (Milner, 1975)

Isomorfismos

Dedução Natural (Gentzen, 1935)	$\longleftrightarrow$	λ-Cálculo Tipado (Church, 1940)
Type Schemes (Hindley, 1969)	$\longleftrightarrow$	Sistema de Tipos da ML (Milner, 1975)
System F (Girard, 1972)	$\longleftrightarrow$	λ-Cálculo Polimórfico (Reynolds, 1974)

Isomorfismos

Dedução Natural (Gentzen, 1935)	$\longleftrightarrow$	λ-Cálculo Tipado (Church, 1940)
Type Schemes (Hindley, 1969)	$\longleftrightarrow$	Sistema de Tipos da ML (Milner, 1975)
System F (Girard, 1972)	$\longleftrightarrow$	λ-Cálculo Polimórfico (Reynolds, 1974)
Lógica Modal (Lewis, 1910)	$\longleftrightarrow$	Mônadas (Kleisi, 1965), (Moggi, 1987)

Isomorfismos

Dedução Natural (Gentzen, 1935)	$\longleftrightarrow$	λ-Cálculo Tipado (Church, 1940)
Type Schemes (Hindley, 1969)	$\longleftrightarrow$	Sistema de Tipos da ML (Milner, 1975)
System F (Girard, 1972)	$\longleftrightarrow$	λ-Cálculo Polimórfico (Reynolds, 1974)
Lógica Modal (Lewis, 1910)	$\longleftrightarrow$	Mônadas (Kleisi, 1965), (Moggi, 1987)
Classical-Intuitionistic Embedding (Gödel, 1933)	$\longleftrightarrow$	Continuation Passing Style (Reynolds, 1972)

A vingança de Reynolds!

Lógica Linear (Girard, 1987)	$\longleftrightarrow$	Syntatic Control of Interference (Reynolds, 1978)
Lógica Linear (Girard, 1987)	$\longleftrightarrow$	Session Types (Honda, 1993)

Coq

Learning resources:

- Software Foundations
- Mathematical Components

Conclusão

PROGRAMS	$\Longleftrightarrow$	LOGIC	$\Longleftrightarrow$	CATEGORIES
type	$\Longleftrightarrow$	formula	$\Longleftrightarrow$	object
function	$\Longleftrightarrow$	implication	$\Longleftrightarrow$	arrow
tuple	$\Longleftrightarrow$	conjunction	$\Longleftrightarrow$	product

Conclusão

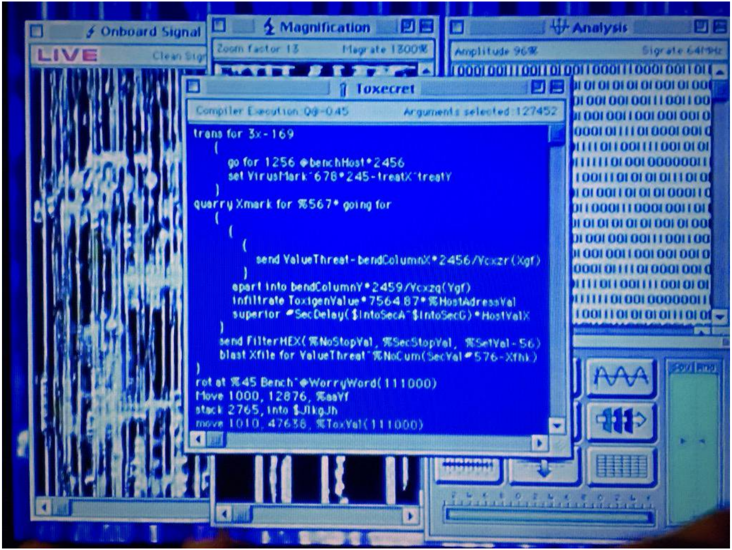


Figura: Independence Day (1996)

Referências I

Philip Wadler (2015)
Propositions as types
Communications of the ACM 58(12), 75 – 84.

Alonzo Church (1936)
An unsolvable problem of elementary number theory
American journal of mathematics 58(2), 345 – 363.

Stephen Cole Kleene (1936)
General recursive functions of natural numbers
Mathematische annalen 112(1), 727 – 742.

Alan M. Turing (1936)
On computable numbers, with an application to the Entscheidungsproblem
Proceedings of the London mathematical society 2(1), 230 – 265.

Girard, Jean-Yves and Taylor, Paul and Lafont, Yves (1989)
Proofs and Types
Cambridge University Press Cambridge 7.

Referências II

Edward Z. Yang, (2012)
Logitext Tutorial
http://logitext.mit.edu/tutorial.