λ-calculus

Valeran MAYTIE

1 Presentation

- 1935 (a theory of computable functions)
  Alonzo Church, attempt at formalizing computation

Functions:

- maths : $f : A \to B$ is a set of pairs
- programming : instruction to compute an output

1.1 Definitions

We can define the set of λ-terms (Λ) with a grammar:

$$\begin{aligned} \Lambda := {} & x, y, z... && \text{(variable)} \\ \mid {} & \lambda.\Lambda && \text{(functions)} \\ \mid {} & \Lambda\ \Lambda && \text{(application)} \end{aligned}$$

The application is left associative: $(l_1\ l_2)\ l_3$.

1.2 Computation

Example, we want to compute $(\lambda xyz.\ x\ z\ (y\ z))\ (\lambda ab.\ a)\ t\ u$

$$\begin{aligned} & (\lambda xyz.\ x\ z\ (y\ z))\ (\lambda ab.\ a)\ t\ u \\ = {} & (\lambda yz.\ (\lambda ab.\ a)\ z\ (y\ z))\ t\ u \\ = {} & (\lambda z.\ (\lambda ab.\ a)\ z\ (t\ z))\ u \\ = {} & (\lambda ab.\ a)\ u\ (t\ u)) \\ = {} & (\lambda b.\ u)\ (t\ u)) \\ = {} & u \end{aligned}$$

Here are some examples of slightly more subtle calculations:

$$\begin{aligned} & (\lambda x.\ (\lambda x.\ x))\ y \\ = {} & \lambda x.\ x \end{aligned}$$

$$\begin{aligned} & (\lambda x.\ (\lambda y.\ x))\ y \\ = {} & \lambda z.\ y \end{aligned}$$

We will define the reduction rewrite rule called β-reduction later.

1.3 Inductive reasoning

We can also define Λ with the smallest set such that :

- $\forall x \in \mathrm{Var}, x \in \Lambda$
- $\forall x \in \mathrm{Var}, \forall t \in \Lambda, \lambda x.t \in \Lambda$
- $\forall t_1 t_2, t_1\ t_2 \in \Lambda$

We define Λ by induction, so we can write induction function.

For example, we can write f_v the function who compute the number of variable in term t and $f_@$ the function who compute the number of application

$$\begin{cases} f_v(x) & = 1 \\ f_v(\lambda x.t) & = f_v(t) \\ f_v(t_1\ t_2) & = f_v(t_1) + f_v(t_2) \end{cases} \qquad \begin{cases} f_@(x) & = 0 \\ f_@(\lambda x.t) & = f_@(t) \\ f_@(t_1\ t_2) & = 1 + f_@(t_1) + f_@(t_2) \end{cases}$$

How to prove that some property $P(t)$ is valid for all λ-terms t ?

1. Prove that $\forall x \in \mathrm{Var}, P(x)$ is valid
2. Prove that $\forall x \in \mathrm{Var}, \forall t, P(t) \Rightarrow P(\lambda x.\ t)$ is valid
3. Prove that $\forall t_1, t_2, P(t_1) \wedge P(t_2) \Rightarrow P(t_1\ t_2)$ is valid

Example 1 – We want to prove $H : \forall t, f_v(t) = 1 + f_@(t)$

Proof – We proof H by induction on the term t :

- $t = x$, $f_v(x) = x$ and $f_@(x) = 0$, so we have $f_v(x) = 1 + f_@(x)$
- $t = \lambda x.t$, we assume that $f_v(t) = 1 + f_@(t)$. We calculate $f_v(\lambda x.t) = f_v(t) = 1 + f_@(t) = 1 + f_@(\lambda x.t)$
- $t = t_1\ t_2$, we assume that $f_v(t_1) = 1 + f_@(t_1)$ and $f_v(t_2) = 1 + f_@(t_2)$. By the calculation $f_v(t_1\ t_2) = f_v(t_1) + f_v(t_2) = 1 + f_@(t_1) + 1 + f_@(t_2) = 1 + f_@(t_1\ t_2)$

□

1.4 Bound variables and free variables

1.5 α-equivalence

1.6 β-reduction

Example 2 – Make them nice

- $\lambda x.\ (\lambda x.\ x\ y)(\lambda y.x\ y)$
- $\lambda xy.\ x(\lambda y.\ (\lambda y.\ y)\ y\ z)$

Example 3 – Compute $(\lambda f.\ f\ f)\ (\lambda ab.b\ a\ b)$

Example 4 – Prove that $fv(t[x \leftarrow u]) \subseteq (fv(t)\ \{x\}) \cup fv(u)$