1 Untyped $\lambda$-calculus

Terms. The set of lambda expressions or terms can be defined inductively.

1. If $x$ is a variable, then $x$ is a lambda expression;
2. If $x$ is a variable and $e$ is a lambda expression, then $\lambda x.e$ is a lambda expression. This is called $\lambda$-abstraction.
3. If $e_1$ and $e_2$ are lambda expressions, then $(e_1 e_2)$ is a lambda expression. This is called application.

$$e ::= x \mid \lambda x.e \mid e_1 e_2$$

Free and bound variables.

$$\text{FV}(x) = \{x\}$$
$$\text{FV}(\lambda x.e) = \text{FV}(e) \setminus \{x\}$$
$$\text{FV}(e_1 e_2) = \text{FV}(e_1) \cup \text{FV}(e_2)$$

Capture-avoiding substitutions.

$$x[e/y] = \begin{cases} 
  e & \text{if } x = y \\
  x & \text{if } x \neq y 
\end{cases}$$

$$(\lambda x.e)[t/y] = \begin{cases} 
  \lambda x.e & \text{if } x = y \\
  \lambda x.(e[t/y]) & \text{if } x \neq y \text{ and } x \not\in \text{FV}(t) 
\end{cases}$$

$$(e_1 e_2)[t/y] = (e_1[t/y] e_2[t/y])$$
**α-conversion.** Also called alpha renaming, or α-equivalence, allows bound variable names to be changed.

\[ \lambda x. e \equiv \lambda y. (e[y/x]) \]

where \( y \) is a fresh variable, or \( y \neq x \) and \( y \notin \text{FV}(e) \).

In a nameless notation called De Bruijn index [1], any two α-equivalent terms are literally identical.

**beta-reduction**  
Beta-reduction is a kind of function application.

\[ (\lambda x. e)e' \beta e'[x/x] \]

**η-conversion**  
η-conversion defines function extensionality: Two functions are extensionally equivalent if and only if they compute the same result for all the inputs.

\[ \lambda x. (fx) \equiv f \quad (x \notin \text{FV}(f)) \]

## 2 Simply typed λ-calculus

**Terms.**

\[ e ::= x | \lambda x : \tau. e | e_1 e_2 \]

**Typing**

\[
\begin{align*}
\frac{x : \tau \in \Gamma}{\Gamma \vdash x : \tau} & \quad \text{TyVar} \\
\frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \lambda x. \tau_1 e : \tau_1 \rightarrow \tau_2} & \quad \text{TyAbs} \\
\frac{\Gamma \vdash e_1 : \tau_1 \rightarrow \tau_2 \quad \Gamma \vdash e_2 : \tau_1}{\Gamma \vdash (e_1 e_2) : \tau_2} & \quad \text{TyApp}
\end{align*}
\]

## 3 Homework 2

The project in Rojas [3].

## 4 Bibliography Notes and Further reading

Chapter 5–7, 9–10 of Pierce [2].

References


