Syntax

$$\begin{aligned}
e &= x \mid \lambda x.e \mid e\,e && \text{(Expressions)}\\
v &= \lambda x.e && \text{(Values)}\\
a &= A[v] && \text{(Answers)}\\
A &= [\,] \mid (\lambda x.A)\,e && \text{(Answer Contexts)}\\
E &= [\,] \mid E\,e \mid (\lambda x.E[x])\,E \mid (\lambda x.E)\,e && \text{(Evaluation Contexts)}
\end{aligned}$$

Notions of Reduction

$$\begin{array}{rcl}
(\lambda x.E[x])\,v & \boldsymbol{\beta}_{\mathbf{need}} & E[x]\{x := v\}\\
(\lambda x.A[v])\,e\,e' & \textbf{assoc-L} & (\lambda x.A[v\,e'])\,e\\
(\lambda x.E[x])\,((\lambda y.A[v])\,e) & \textbf{assoc-R} & (\lambda y.A[(\lambda x.E[x])\,v])\,e
\end{array}$$

$$\begin{aligned}
&\mathbf{need} = \boldsymbol{\beta}_{\mathbf{need}} \cup \textbf{assoc-L} \cup \textbf{assoc-R}\\
&\to : \text{compatible closure of } \mathbf{need}\\
&\twoheadrightarrow : \text{reflexive, transitive closure of } \to\\
&\Rightarrow : \text{parallel extension of } \to\\
&\mapsto : \text{standard reduction relation}\\
&\quad E[e] \mapsto E[e'] \text{ if } e\,\mathbf{need}\,e'\\
&\mapsto\!\!\!\!\rightarrow : \text{reflexive, transitive closure of } \mapsto
\end{aligned}$$

Definition 1 ($\Rightarrow$)**.** *Parallel reduction of all non-overlapping redexes.*

$$\begin{array}{rcll r}
e & \Rightarrow & e & & (1)\\
(\lambda x.E[x])\,v & \Rightarrow & E'[x]\{x := v'\} & \text{if } E[x] \Rightarrow E'[x],\ v \Rightarrow v' & (2)\\
(\lambda x.A[v])\,e_1\,e_2 & \Rightarrow & (\lambda x.A'[v'\,e_2'])\,e_1' & \text{if } A[v] \Rightarrow A'[v'],\ e_1 \Rightarrow e_1',\ e_2 \Rightarrow e_2' & (3)\\
(\lambda x.E[x])\,((\lambda y.A[v])\,e) & \Rightarrow & (\lambda y.A'[(\lambda x.E'[x])\,v'])\,e' & \text{if } E[x] \Rightarrow E'[x],\ A[v] \Rightarrow A'[v'],\ e \Rightarrow e' & (4)\\
e_1\,e_2 & \Rightarrow & e_1'\,e_2' & \text{if } e_1 \Rightarrow e_1',\ e_2 \Rightarrow e_2' & (5)\\
\lambda x.e & \Rightarrow & \lambda x.e' & \text{if } e \Rightarrow e' & (6)
\end{array}$$

Definition 2 ($\diamond$)**.** *Standard reduction sequences* $\mathcal{R}$

- $x \subset \mathcal{R}$
- *If* $e_1 \diamond \cdots \diamond e_m \in \mathcal{R}$, *then* $\lambda x.e_1 \diamond \cdots \diamond \lambda x.e_m \in \mathcal{R}$
- *If* $e_1 \diamond \cdots \diamond e_m \in \mathcal{R}$ *and* $e_1' \diamond \cdots \diamond e_n' \in \mathcal{R}$ *then* $(e_1\,e_1') \diamond \cdots \diamond (e_m\,e_1') \diamond (e_m\,e_2') \diamond \cdots \diamond (e_m\,e_n') \in \mathcal{R}$
- *If* $e_1 \diamond \cdots \diamond e_m \in \mathcal{R}$ *and* $e_0 \mapsto e_1$, *then* $e_0 \diamond e_1 \diamond \cdots \diamond e_m \in \mathcal{R}$

Definition 3 (size function $|\cdot|$)**.**

$$\begin{aligned}
|e \Rightarrow e| &= 0\\
|(\lambda x.e)\,v \Rightarrow e'\{x := v'\}| &= |e \Rightarrow e'| + \#(x, e') \times |v \Rightarrow v'| + 1\\
|(\lambda x.A[v])\,e_1\,e_2 \Rightarrow (\lambda x.A'[v'\,e_2'])\,e_1'| &= |A[v] \Rightarrow A'[v']| + |e_1 \Rightarrow e_1'| + |e_2 \Rightarrow e_2'| + 1\\
|(\lambda x.E[x])\,((\lambda y.A[v])\,e) \Rightarrow (\lambda y.A'[(\lambda x.E'[x])\,v'])\,e'| &= |A[v] \Rightarrow A'[v']| + |E[x] \Rightarrow E'[x]| + |e \Rightarrow e'| + 1\\
|(e_1\,e_2) \Rightarrow (e_1'\,e_2')| &= |e_1 \Rightarrow e_1'| + |e_2 \Rightarrow e_2'|\\
|\lambda x.e \Rightarrow \lambda x.e'| &= |e \Rightarrow e'|
\end{aligned}$$

Theorem 1 (Curry-Feys Standardization). *$e \twoheadrightarrow e'$ iff $e_1 \diamond \cdots \diamond e_n \in \mathcal{R}$ s.t. $e = e_1$ and $e' = e_n$.*

Proof. Use lemma 1 □

Lemma 1. *If $e \Rightarrow e'$ and $e' \diamond e'_1 \diamond \cdots \diamond e'_n \in \mathcal{R}$, $\exists e_1 \diamond e_2 \diamond \cdots \diamond e_p \in \mathcal{R}$ s.t. $e_1 = e$ and $e_p = e'_n$.*

Proof. By triple lexicographic induction on 1) length n of given standard reduction sequence, 2) $|e \Rightarrow e'|$, and 3) structure of e. Proceed by case analysis on last step in derivation of $e \Rightarrow e'$

Case $e = e'$**.** (base)

Then $e_1 = e = e'$ and $e_p = e'_n$

Case $e \Rightarrow e'$ by $\Rightarrow def(2)$**.**

$e = (\lambda x.E[x])\, v$, $e' = E'[x]\{x := v'\}$, $E[x] \Rightarrow E'[x]$, $v \Rightarrow v'$
Let $e_3 = E[x]\{x := v\}$. Then $e \mapsto e_3$.
Since $e_3 \Rightarrow e'$ (by subst lemma) and $|e_3 \Rightarrow e'| < |e \Rightarrow e'|$ (by lemma 4),
then by IH $\exists e_4 \diamond \cdots \diamond e_p \in \mathcal{R}$ s.t. $e_4 = e_3$ and $e_p = e'_n$.
Thus $e \diamond e_4 \diamond \cdots \diamond e_p \in \mathcal{R}$ is the required sr sequence.

Case $e \Rightarrow e'$ by $\Rightarrow def(3)$**.**

$e = (\lambda x.A[v])\, e_3\, e_4$, $e' = (\lambda x.A'[v'\, e'_4])\, e'_3$, $A[v] \Rightarrow A'[v']$, $e_3 \Rightarrow e'_3$, $e_4 \Rightarrow e'_4$
Let $e_5 = (\lambda x.A[v\, e_4])\, e_3$. Then $e \mapsto e_5$.
Since $e_5 \Rightarrow e'$ (by $\Rightarrow def(5)$) and $|e_5 \Rightarrow e'| < |e \Rightarrow e'|$ (by lemma 4),
then by IH, $\exists e_6 \diamond \cdots \diamond e_p \in \mathcal{R}$ s.t. $e_6 = e_5$ and $e_p = e'_n$.
Thus $e \diamond e_6 \diamond \cdots \diamond e_p \in \mathcal{R}$ is the required sr sequence.

Case $e \Rightarrow e'$ by $\Rightarrow def(4)$**.**

$e = (\lambda x.E[x])\, ((\lambda y.A[v])\, e_3)$, $e' = (\lambda y.A'[(\lambda x.E'[x])\, v'])\, e'_3$, $E[x] \Rightarrow E'[x]$, $A[v] \Rightarrow A'[v']$, $e_3 \Rightarrow e'_3$
Let $e_4 = (\lambda y.A[(\lambda x.E[x])\, v])\, e_3$. Then $e \mapsto e_4$.
Since $e_4 \Rightarrow e'$ (by $\Rightarrow def(5)$) and $|e_4 \Rightarrow e'| < |e \Rightarrow e'|$ (by lemma 4),
then by IH, $\exists e_5 \diamond \cdots \diamond e_p \in \mathcal{R}$ s.t. $e_4 = e_3$ and $e_p = e'_n$.
Thus $e \diamond e_5 \diamond \cdots \diamond e_p \in \mathcal{R}$ is the required sr sequence.

Case $e \Rightarrow e'$ by $\Rightarrow def(5)$**.**

$e = e_3\, e_4$, $e' = e'_3\, e'_4$, $e_3 \Rightarrow e'_3$, $e_4 \Rightarrow e'_4$

Subcase $e_1 \diamond e_2 = e_1 \mapsto e_2$**.**

Subcase *otherwise***.**

Both subcases follow same reasoning as text.

Case $e \Rightarrow e'$ by $\Rightarrow def(6)$**.**

Claim holds by IH.

□

Lemma 2. *If $e \Rightarrow e^* \mapsto e^{**}$, then $\exists e^{***}$ s.t. $e \mapsto\!\!\!\!\!\!\;\rightarrow e^{***} \Rightarrow e^{**}$.*

Proof. By double lexicographic induction on $|e \Rightarrow e^*|$ and structure of e.

Case $e = e^*$**.**

$e^{***} = e^{**}$

Case $e = (\lambda x.E[x])\, v, e^* = E'[x]\{x := v'\}, E[x] \Rightarrow E'[x], v \Rightarrow v'$**.**

Case $e = (\lambda x.A[v])\, e_1\, e_2, e^* = (\lambda x.A'[v'\, e'_2])\, e'_1, A[v] \Rightarrow A'[v'], e_1 \Rightarrow e'_1, e_2 \Rightarrow e'_2$**.**

Case $e = (\lambda x.E[x])\, ((\lambda y.A[v])\, e_1), e^* = (\lambda y.A'[(\lambda x.E'[x])\, v'])\, e'_1, E[x] \Rightarrow E'[x], A[v] \Rightarrow A'[v'], e_1 \Rightarrow e'_1$**.**

Each of these cases follows from IH on size and lemma 4.

Case $e = e_1\, e_2, e^* = e_1'\, e_2', e_1 \Rightarrow e_1', e_2 \Rightarrow e_2'$**.**

Subcases of $e^* \mapsto e^{**}$:

Subcase $e_1' = \lambda x.E[x], e_2' = v, e^{**} = E[x]\{x := v\}$**.**

Subcase $e_1' = (\lambda x.A[v])\, e_3, e_2' = e_4, e^{**} = (\lambda x.A[v\, e_4])\, e_3$**.**

Subcase $e_1' = \lambda x.E[x], e_2' = (\lambda y.A[v])\, e_3, e^{**} = (\lambda y.A[\lambda x.E[x]\, v])\, e_3$**.**

Each of these cases follows from IH on subterms and lemma 3

Subcase $e^* = E[e_3]\, e_2', e^{**} = E[e_3']\, e_2', e_3$ **need** e_3'**.**

Subcase $e^* = e_1'\, E[e_3], e^{**} = e_1'\, E[e_3'], e_3$ **need** e_2'**.**

Each of these cases follows by IH on subterms.

Case $e = \lambda x.e_1, e^* = \lambda x.e_1'$**.**

Impossible bc SR undefined for values.

$\square$

Lemma 3.

1. *If* $e \Rightarrow \lambda x.E[x]$*,* $\exists \lambda x.E'[x]$ *s.t.* $e \mapsto\!\!\!\!\to \lambda x.E'[x] \Rightarrow \lambda x.E[x]$*.*
2. *If* $e \Rightarrow a$*,* $\exists a'$ *s.t.* $e \mapsto\!\!\!\!\to a' \Rightarrow a$*.*
3. *If* $e \Rightarrow x$*, then* $e \mapsto\!\!\!\!\to x$*.*

Lemma 4 (Size)**.** *If* $s_{E[x]} = |E[x] \Rightarrow E'[x]|$*,* $s_v = |v \Rightarrow v'|$*,* $s_{A[v]} = |A[v] \Rightarrow A'[v']|$*,* $s_{e_1} = |e_1 \Rightarrow e_1'|$*,* $s_{e_2} = |e_2 \Rightarrow e_2'|$ *then:*

1. $E[x]\{x := v\} \Rightarrow E'[x]\{x := v'\}$
2. $|E[x]\{x := v\} \Rightarrow E'[x]\{x := v'\}| \leq s_{E[x]} + \#(x, E'[x]) \times s_v$
3. $(\lambda x.A[v\, e_2])\, e_1 \Rightarrow (\lambda x.A'[v'\, e_2'])\, e_1'$
4. $|(\lambda x.A[v\, e_2])\, e_1 \Rightarrow (\lambda x.A'[v'\, e_2'])\, e_1'| \leq s_{A[v]} + s_{e_1} + s_{e_2}$
5. $(\lambda y.A[(\lambda x.E[x])\, v])\, e_1 \Rightarrow (\lambda y.A'[(\lambda x.E'[x])\, v'])\, e_1'$
6. $|(\lambda y.A[(\lambda x.E[x])\, v])\, e_1 \Rightarrow (\lambda y.A'[(\lambda x.E'[x])\, v'])\, e_1'| \leq s_{A[v]} + s_{E[x]} + s_{e_1}$