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

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

$$
\begin{aligned}
\mathbf{need} &= \boldsymbol{\beta}_{\mathbf{need}} \cup \textbf{assoc-L} \cup \textbf{assoc-R}\\
&\rightarrow : \text{compatible closure of } \mathbf{need}\\
&\twoheadrightarrow : \text{reflexive, transitive closure of } \rightarrow\\
&\Rightarrow : \text{parallel extension of } \rightarrow
\end{aligned}
$$

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

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

Theorem 1 (Church-Rosser)**.** *If* $e \twoheadrightarrow e_1$ *and* $e \twoheadrightarrow e_2$, $\exists e'$ *s.t.* $e_1 \twoheadrightarrow e'$ *and* $e_2 \twoheadrightarrow e'$.

Proof. Use Diamond Property of $\Rightarrow$. □

Lemma 1 (Diamond Property of $\Rightarrow$)**.** *If* $e \Rightarrow e_1$ *and* $e \Rightarrow e_2$, $\exists e'$ *s.t.* $e_1 \Rightarrow e'$ *and* $e_2 \Rightarrow e'$.

Proof. By structural induction on proof of $e \Rightarrow e_1$. All cases make use of lemmas 2 and 3.

Case $e = e_1$**.** (base)

Then $e_1 \Rightarrow e_2$, so $e' = e_2$.

Case $e \Rightarrow e_1$ by $\Rightarrow def(2)$**.**

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(2)$**.**

- $e_1 \Rightarrow e'$ by substitution lemma
- $e_2 \Rightarrow e'$ by substitution lemma

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(5)$**.**

- $e_1 \Rightarrow e'$ by substitution lemma
- $e_2 \Rightarrow e'$ by $\Rightarrow def(2)$

Case $e \Rightarrow e_1$ by $\Rightarrow def(3)$.

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(3)$.

- $e_1 \Rightarrow e'$ by $\Rightarrow def(5)$
- $e_2 \Rightarrow e'$ by $\Rightarrow def(5)$

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(5)$.

- $e_1 \Rightarrow e'$ by $\Rightarrow def(5)$
- $e_2 \Rightarrow e'$ by $\Rightarrow def(3)$

Case $e \Rightarrow e_1$ by $\Rightarrow def(4)$.

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(4)$.

- $e_1 \Rightarrow e'$ by $\Rightarrow def(5)$
- $e_2 \Rightarrow e'$ by $\Rightarrow def(5)$

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(5)$.

- $e_1 \Rightarrow e'$ by $\Rightarrow def(5)$
- $e_2 \Rightarrow e'$ by $\Rightarrow def(4)$

Case $e \Rightarrow e_1$ by $\Rightarrow def(5)$.

$e \Rightarrow e_2$ subcases by $\Rightarrow def(2),(3),(4),(5)$, analogous to above cases.

Case $e \Rightarrow e_1$ by $\Rightarrow def(6)$.

Claim holds by IH.

□

Lemma 2 ($E[x]$ closed under $\Rightarrow$). *If $e = E[x]$, $x \in fv(E[x])$, and $e \Rightarrow e'$, then for some E', $e' = E'[x]$, $x \in fv(E'[x])$.*

Proof. Structural induction on E.

Case $E = [\,]$.

$e = e' = x$, $E' = [\,]$.

Case $E = E_1\, e_1$.

IH: $E_1[x] \Rightarrow E_1'[x]$, so $E_1[x]\, e_1 \Rightarrow E_1'[x]\, e_1'$.
$E_1'[x]\, e_1'$ is not **need** redex bc $E_1'[x] \notin v$, by lemma 4, so $E' = E_1'\, e_1'$,

Case $E = (\lambda y.E_2[y])\, E_1$.

IH: $E_2[y] \Rightarrow E_2'[y]$, $E_1[x] \Rightarrow E_1'[x]$, so $(\lambda y.E_2[y])\, E_1[x] \Rightarrow (\lambda y.E_2'[y])\, E_1'[x]$.
$(\lambda y.E_2'[y])\, E_1'[x]$ is not $\boldsymbol{\beta}_{\mathbf{need}}$ redex bc $E_1'[x] \notin v$, by lemma 4.
$(\lambda y.E_2'[y])\, E_1'[x]$ is not **assoc-L** redex bc $E_2'[y] \notin a$, by lemma 4.
$(\lambda y.E_2'[y])\, E_1'[x]$ is not **assoc-R** redex bc $E_1'[x] \notin a$, by lemma 4.
so $E' = (\lambda y.E_2'[y])\, E_1'$,

Case $E = (\lambda y.E_1)\, e_1$.

IH: $E_1[x] \Rightarrow E_1'[x]$, so $(\lambda y.E_1[x])\, e_1 \Rightarrow (\lambda y.E_1'[x])\, e_1'$
$(\lambda y.E_1'[x])\, e_1'$ is not $\boldsymbol{\beta}_{\mathbf{need}}$ redex bc $E_1'[x] \neq E_2[y]$.
$(\lambda y.E_1'[x])\, e_1'$ is not **assoc-L** redex bc $E_1'[x] \notin a$, by lemma 4.
$(\lambda y.E_1'[x])\, e_1'$ is not **assoc-R** redex bc $E_1'[x] \neq E_2[y]$.
so $E' = (\lambda y.E_1')\, e_1'$,

□

Lemma 3 ($A[v]$ closed under $\Rightarrow$)**.** *If $e = A[v]$ and $e \Rightarrow e'$, then $e' = A'[v']$.*

Proof. Structural induction on A.

Case $A = [\,]$**.**

$e' = v'$

Case $A = (\lambda x.A_1)\, e_1$**.**

IH: $A_1[v] \Rightarrow A_1'[v']$, so $(\lambda x.A_1[v])\, e_1 \Rightarrow (\lambda x.A_1'[v'])\, e_1'$.
$(\lambda x.A_1'[v'])\, e_1'$ is not $\boldsymbol{\beta}_{\mathbf{need}}$ redex bc $A_1'[v'] \neq E[x]$, by lemma 5.
$(\lambda x.A_1'[v'])\, e_1'$ is not **assoc-L** redex bc e was not.
$(\lambda x.A_1'[v'])\, e_1'$ is not **assoc-R** redex bc $A_1'[v'] \neq E[x]$, by lemma 5.
so $e' = (\lambda x.A_1'[v'])\, e_1'$,

□

Lemma 4. *If $e = E[x]$, $e \notin a$.*

Proof. By structural induction on E.

Case $E = [\,]$**.**

Case $E = E_1\, e_1$**.**

IH: $E_1[x] \notin a$, so $E_1[x]$ is not a λ.

Case $E = (\lambda y.E_2[y])\, E_1$**.**

IH: $E_2[y] \notin a$, $E_1[x] \notin a$,

Case $E = (\lambda y.E_1)\, e_1$**.**

IH: $E_1[x] \notin a$

□

Lemma 5. *If $e = a$, then $\nexists E$ s.t. $e = E[x]$.*

Lemma 6. *If $e = E[x]$, $x \in fv(E[x])$, $\nexists E', y \neq x$, $y \in fv(E')$ s.t. $e = E'[y]$.*

Lemma 7. *If $e = E[x]$, $x \in fv(E[x])$, $E[x]$ is not a* ***need*** *redex.*