Adaptive Analysis

April 7, 2020

Types	$\tau ::= b \mid \tau \multimap \tau' \mid !_n\tau \mid \tau \times \tau \mid \forall i{::}\mathbb{N}.\tau \mid \mathrm{Q}$
Term	$t ::= c \mid \mathsf{fix} f(x).t \mid\ t\ t\ \mid !t \mid (t_1, t_2) \mid\ \mathsf{let}\ !x = t_1\ \mathsf{in}\ t_2\ \mid \Lambda.t \mid t[] \mid\ \lambda x.t\ \mid M(t) \mid x \mid q \mid$ $\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} \mid\ \mathsf{let}\ (x_1, x_2) = t_1\ \mathsf{in}\ t_2$
Normal Form	$v ::= c \mid \mathsf{fix} f(x).t \mid !t \mid (v_1, v_2) \mid \Lambda.t \mid\ \lambda x.t\ \mid x \mid q \mid \mathsf{case}\ v\ \mathsf{of}\ \{c_i \Rightarrow v_i\}_{c_i \in b_i} \mid$ $\mathrm{nil} \mid \mathrm{cons}(v_1, v_2)$
Mechanisms	$M ::= \mathtt{gauss} \mid \mathtt{thdt}$
Tree	$T_b ::= c \mid M(T_{query}) \mid \mathsf{case}\ T_b\ \mathsf{of}\ \{c_i \Rightarrow T_{b_i}\}_{c_i \in b}$ $T_{query} ::= q \mid \mathsf{case}\ T_b\ \mathsf{of}\ \{c_i \Rightarrow T_{query_i}\}_{c_i \in b}$
Depth	$\mathsf{depth}(c) = 0$ $\mathsf{depth}(!t) = \mathsf{depth}(t)$ $\mathsf{depth}(t_1\ t_2) = \max(\mathsf{depth}(t_1), \mathsf{depth}(t_2))$ $\mathsf{depth}(M(t)) = 1 + \mathsf{depth}(t)$ $\mathsf{depth}(\lambda x.t) = \mathsf{depth}(t)$ $\mathsf{depth}(x) = 0$ $\mathsf{depth}(q) = 0$ $\mathsf{depth}((t_1, t_2)) = \max(\mathsf{depth}(t_1), \mathsf{depth}(t_2))$ $\mathsf{depth}(\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t') = \max(\mathsf{depth}(t), \mathsf{depth}(t'))$ $\mathsf{depth}(\mathsf{let}\ !x = t\ \mathsf{in}\ t') = \max(\mathsf{depth}(t), \mathsf{depth}(t'))$ $\mathsf{depth}(\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b}) = \max(\mathsf{depth}(t), \mathsf{depth}(t_i))$ $\mathsf{depth}(\Lambda.t) = \mathsf{depth}(t)$ $\mathsf{depth}(t\,[]) = \mathsf{depth}(t)$

Figure 1: syntax

Abstract example

$$\texttt{gauss}(\texttt{count}(\phi))$$

Concrete example

$$\texttt{gauss}(\texttt{count}(\lambda \texttt{r}.\pi_1 \texttt{r} \leq 5))$$

$$\texttt{gauss}(\texttt{count}(\lambda \texttt{r}.\pi_1 \texttt{r} \leq 0.134)) + \texttt{gauss}(\texttt{count}(\lambda \texttt{r}.\pi_2 \texttt{r} = \texttt{"hiv"}))$$

Depth 2
Abstract

$$\mathsf{case}\ \texttt{gauss}(\texttt{count}(\phi))\ \mathsf{of}\ \{c_i \Rightarrow \texttt{gauss}(\texttt{count}(\phi_{\texttt{i}}))\}_{c_i \in b}$$

Concrete

$$\phi = \lambda r.\pi_1 r \leq 5; c_1 = 0, \phi_1 = \lambda r.\pi_1 r \leq 5; c_2 = 0.1, \phi_2 = \lambda r.\pi_1 r \leq 3; \cdots$$

Depth 3:

$\mathsf{case}\ \texttt{gauss}(\texttt{count}(\phi))\ \mathsf{of}$
$\{c_i \Rightarrow \mathsf{case}\ \texttt{gauss}(\texttt{count}(\phi'))\ \mathsf{of}$
$\{c_i' \Rightarrow \texttt{gauss}(\texttt{count}(\phi'_{\texttt{i}}))\}_{c_i' \in b}\}_{c_i \in b}$

Figure 2: simple examples

Algorithm 1 A two-round analyst strategy for random data (Algorithm 4 in ...)

Require: Mechanism $\mathcal{M}$ with a hidden state $X \in \{-1,+1\}^{n\times(k+1)}$.
  for $j \in [k]$ **do**.
    define $q_j(x) = x(j) \cdot x(k)$ where $x \in \{-1,+1\}^{k+1}$.
    let $a_j = \mathcal{M}(q_j)$
    {In the line above, $\mathcal{M}$ computes approx. the exp. value of q_j over X. So, $a_j \in [-1,+1]$.}
  define $q_{k+1}(x) = \mathrm{sign}\left(\sum_{i\in[k]} x(i) \times \ln \frac{1+a_i}{1-a_i}\right)$ where $x \in \{-1,+1\}^{k+1}$.
  {In the line above, $\mathrm{sign}(y) = \begin{cases} +1 & \text{if } y \geq 0 \\ -1 & \text{otherwise} \end{cases}$.}
  let $a_{k+1} = \mathcal{M}(q_{k+1})$
  {In the line above, $\mathcal{M}$ computes approx. the exp. value of q_{k+1} over X. So, $a_{k+1} \in [-1,+1]$.}
  return a_{k+1}.
Ensure: $a_{k+1} \in [-1,+1]$

Two-rounds:

```
let !g = !(fix f(j).λk.
  if (j < k) then
       let a = M (λx.(x j) · (x k)) in
           (a, j) :: (f (j + 1) k)
           else [])
  in
let !l = !g 0 K in
let !q = !λx.sign (foldl (λacc.λ(a, i).(acc + (x i) ∗ lg((1+a)/(1−a))) 0 l)) in
M(q)
```

$x : \text{int} \to \text{int}$
$\cdot : \text{int} * \text{int} \to \text{int}$
$g : \text{int} \to \text{int} \to b * \text{intlist}$
$q : \text{query}$
$M : \text{query} \to b$

Type derivation:

Let $A = \big(\text{int} \to \text{int} \to b * \text{intlist}\big)$, $\Gamma = f : A, M : \text{query} \to b, j : \text{int}, k : \text{int}$, $\Gamma_0 = M : \text{query} \to b$, $[\Delta]_i = g : [A]_i, l : [b * \text{intlist}]_i, q : [\text{query}]_i$.

$$\frac{\Pi_L \rhd M : \text{query} \to b \vdash_2 !\mathsf{fix}\, f \cdots : !_1 A \qquad \Pi_R \rhd M : \text{query} \to b, g : [A]_1 \vdash_2 \mathsf{let}\ !l = !g\, 0\, K \ \mathsf{in}\ \mathsf{let}\ !q = \cdots \ \mathsf{in}\ \cdots : b}{M : \text{query} \to b \vdash_2 \mathsf{let}\ !g = !\mathsf{fix}\, f \cdots \ \mathsf{in}\ \mathsf{let}\ !l = \cdots \ \mathsf{in}\ \cdots : b}\ \text{LET-B}$$

Derivation Π_L and Π_R are shown as follows:
Π_L:

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{}{\Gamma \vdash_0 j < k : \text{bool}}\ \text{BOOL} \qquad \cfrac{\cdots}{\Gamma \vdash_1 \mathsf{let}\ a = M(\cdots)\ \mathsf{in}\ (a, j) :: \cdots : b * \text{intlist}}\ \text{LET} \qquad \cfrac{}{\Gamma \vdash_0 [] : b * \text{intlist}}\ \text{NIL}}{f : A, M : \text{query} \to b, j : \text{int}, k : \text{int} \vdash_1 \mathsf{if} \cdots : b * \text{intlist}}\ \text{IF}}{f : A, M : \text{query} \to b \vdash_1 \lambda j.\lambda k.\mathsf{if} \cdots : \text{int} \to \text{int} \to b * \text{intlist}}\ \text{ABS}}{M : \text{query} \to b \vdash_1 \mathsf{fix}\, f \cdots : A}\ \text{FIX}}{M : \text{query} \to b \vdash_2 !\mathsf{fix}\, f \cdots : !_1 A}\ \text{PR}$$

$$\cfrac{\cfrac{\cfrac{}{\Gamma \vdash_0 \lambda x.(x\, j) \cdot (x\, k) : \text{query}}\ \text{QUERY}}{\Gamma \vdash_1 M(\lambda x.(x\, j) \cdot (x\, k)) : b}\ \text{MT} \qquad \cfrac{\cfrac{}{\Gamma, a : b \vdash_0 (a, j) : b * \text{int}}\ \text{VAR} \qquad \cfrac{\cdots}{\Gamma, a : b \vdash_0 (f\ j + 1\ k) : b * \text{intlist}}\ \text{APP}}{\Gamma, a : b \vdash_0 (a, j) :: (f\ j + 1\ k) : b * \text{intlist}}\ \text{CONS}}{\Gamma \vdash_1 \mathsf{let}\ a = M(\lambda x.(x\, j) \cdot (x\, k))\ \mathsf{in}\ (a, j) :: (f\ j + 1\ k) : b * \text{intlist}}\ \text{LET}$$

Π_R:

$$\cfrac{\cfrac{\cfrac{\cfrac{\cdots}{g : A \vdash_0 g\, 0\, K : b * \text{intlist}}\ \text{APP}}{g : [A]_0 \vdash_0 g\, 0\, K : b * \text{intlist}}\ \text{DER}}{g : [A]_1 \vdash_1 !g\, 0\, K : !_1 b * \text{intlist}}\ \text{PR} \qquad \cfrac{\cfrac{\cfrac{\cfrac{}{x : row \vdash_0 sign \ldots : b}\ \text{VAR}}{\vdash_0 \lambda x. \ldots : row \to b}\ \text{ABS}}{\vdash_1 !\lambda x. \ldots : !_1 row \to b}\ \text{PR} \qquad \cfrac{\cfrac{}{\Gamma_0, \Delta_0 \vdash_0 q : \text{query}}\ \text{QUERY}}{\Gamma_0, [\Delta]_1 \vdash_1 M(q) : b}\ \text{MT}}{\Gamma_0, g : [A]_1, l : [b * \text{intlist}]_1 \vdash_1 \mathsf{let}\ !q = \cdots \ \mathsf{in}\ \cdots : b}\ \text{LET-B}}{\Gamma_0, g : [A]_1 \vdash_1 \mathsf{let}\ !l = !g\, 0\, K\ \mathsf{in}\ \mathsf{let}\ !q = \ldots\ \mathsf{in}\ \ldots : b}\ \text{LET-B}$$

Figure 3: examples: two rounds

Evaluation:

$$\dfrac{\dfrac{}{!\mathsf{fix}\,f\cdots \Downarrow^0 !\mathsf{fix}\,f\cdots}\ \text{E-BANG} \qquad \dfrac{}{\mathsf{fix}\,f\cdots\Downarrow^0 \mathsf{fix}\,f\cdots}\ \text{E-FIX} \qquad \dfrac{\cdots}{\mathsf{let}\ !l = !(\mathsf{fix}\ldots 0K)\ \mathsf{in}\ \cdots \Downarrow^2 c}\ \text{E-LET-BANG}}{\mathsf{let}\ !g = !\mathsf{fix}\,f\cdots\ \mathsf{in}\ \mathsf{let}\ !l=\cdots\ \mathsf{in}\ \cdots\Downarrow^2 c}\ \text{E-LET-BANG}$$

$$\dfrac{\dfrac{}{!(\mathsf{fix}\ldots 0K)\Downarrow^0 !(\mathsf{fix}\ldots 0K)}\ \text{E-BANG} \qquad \dfrac{\Pi_1}{\mathsf{fix}\ldots 0K\Downarrow^1 c_1}\ \text{E-APP} \qquad \dfrac{\Pi_2}{\mathsf{let}\ !q = !(!\lambda x.(\ldots 0 c_1))\ \mathsf{in}\ M(q)\Downarrow^1 c}\ \text{E-LET-BANG}}{\mathsf{let}\ !l = !(\mathsf{fix}\ldots 0K)\ \mathsf{in}\ \cdots\Downarrow^2 c}\ \text{E-LET-BANG}$$

$$\Pi_1: \quad \dfrac{\dfrac{}{\mathsf{fix}\ldots\Downarrow^0 \lambda j.\lambda k.\ldots}\ \text{E-FIX} \qquad \dfrac{}{0\,K\Downarrow^{0\,K} 0}\ \text{E-VALUES} \qquad \dfrac{\Pi_{1-1}}{\mathsf{if}\cdots\mathsf{let}\ a = M(\ldots)\ \mathsf{in}\ \ldots\Downarrow^1 c_1}\ \text{E-IF}}{\mathsf{fix}\ldots 0\,K\Downarrow^1 c_1}\ \text{E-APP}$$

$$\Pi_{1-1}: \quad \dfrac{\dfrac{\dfrac{}{M(\ldots)\Downarrow^1 c_1'}\ \text{E-MT} \qquad \dfrac{}{(c_1', j) :: (f\,(j+1)K)\Downarrow^0 c_1}\ \text{E-CONS}}{\mathsf{let}\ a = M(\ldots)\ \mathsf{in}\ \ldots\Downarrow^1 c_1}\ \text{E-LET} \qquad \dfrac{}{[]\Downarrow^0 []}\ \text{E-NIL}}{\mathsf{if}\cdots\mathsf{let}\ a = M(\ldots)\ \mathsf{in}\ \ldots\Downarrow^1 c_1}\ \text{E-IF}$$

E-LET-BANG

$$\Pi_2: \quad \dfrac{\dfrac{}{!(\lambda x.(\ldots 0\,c_1))\Downarrow^0 !(\lambda x.(\ldots 0\,c_1))}\ \text{E-BANG} \qquad \dfrac{}{\lambda x.(\ldots 0\,c_1)\Downarrow^0 q}\ \text{E-QUERY} \qquad \dfrac{}{M(q)\Downarrow^1 c}\ \text{E-MT}}{\mathsf{let}\ !q = !(\lambda x.(\ldots 0\,c_1))\ \mathsf{in}\ M(q)\Downarrow^1 c}$$

Figure 4: examples: two rounds - evaluation

Multi-rounds:

$$
\begin{array}{l}
\mathsf{let}\,\mathsf{multi\_round} = \mathsf{fix}\, f\,(j).\lambda k.\lambda D.\lambda I.\lambda S.\lambda SC. \\
\quad \mathsf{if}\,(j < k)\,\mathsf{then} \\
\qquad\quad \mathsf{let}\, P_j = \mathsf{uniform}(0,1)\,\mathsf{in} \\
\qquad\qquad \mathsf{let} q_j = \lambda x.\mathsf{if}(\mathsf{uniform}(0,1) < p_j)\,\mathsf{then}\,1\,\mathsf{else}\,0\,\mathsf{in} \\
\qquad\qquad \mathsf{let} q_{j,c} = \lambda x.\mathsf{if}(\mathsf{uniform}(0,1) < p_j)\,\mathsf{then}\,1\,\mathsf{else}\,0\,\mathsf{in} \\
\qquad\qquad\quad \mathsf{let} q_{jd} = \mathsf{in} \\
\qquad\qquad \mathsf{let}\, a_j = M(q_j)\,\mathsf{in} \\
\qquad\qquad\; \mathsf{let}\, S_{j,i} = \mathsf{updt1}\, S_{j,c}\, a_j\, p_j\, q_j \mathsf{in} \\
\qquad\qquad\; \mathsf{let}\, S_{j,c} = \mathsf{updt2}\, S_{j,c}\, a_j\, p_j\, q_{j,c}\,\mathsf{in} \\
\qquad\qquad\quad \mathsf{let}\, I_j = \mathsf{updt3}\, S\, SC\,\mathsf{in} \\
f\,(j+1)\,k,(D \setminus I_j)\, I_j\, S_{j,i}\, S_{j,c}
\end{array}
$$

Figure 5: Caption

Types $\quad \tau ::= \tau_1 + \tau_2 \mid \tau\,\mathrm{list}$

Term $\quad t ::= \mathrm{inl}\ \ t \mid \mathrm{inr}\ \ t \mid \mathrm{nil} \mid \mathrm{cons}(t_1, t_2) \mid\ \mathsf{let}\ x = t_1\ \mathsf{in}\ t_2\ \mid\ \mathrm{case}\ (t, x.t_1, y.t_2)$

$\boxed{\Gamma \vdash_{n,m} t : \tau}$

$$\frac{\Gamma, f : \tau_1 \multimap \tau_2, x : \tau_1 \vdash_n t : \tau_2}{\Gamma \vdash_n \mathsf{fix}\, f(x).t : \tau_1 \multimap \tau_2}\ \textsc{fix} \qquad \frac{\Gamma_1 \vdash_{n_1} t : \tau \qquad \Gamma_2, x : \tau \vdash_{n_2} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ x = t\ \mathsf{in}\ t' : \tau'}\ \textsc{let}$$

$$\frac{\Gamma \vdash_{n_1} t : \tau_1}{\Gamma \vdash_{n_1} \mathrm{inl}\ \ t : \tau_1 + \tau_2}\ \textsc{inl} \qquad \frac{\Gamma \vdash_{n_2} t : \tau_2}{\Gamma \vdash_{n_2} \mathrm{inr}\ \ t : \tau_1 + \tau_2}\ \textsc{inr}$$

$$\frac{\Gamma_1 \vdash_{n_1} t : \tau_1 + \tau_2 \qquad \Gamma_2, x : \tau_1 \vdash_{n_2} t_1 : \tau \qquad \Gamma_3, y : \tau_2 \vdash_{n_3} t_2 : \tau}{\max(\Gamma_1, \Gamma_2, \Gamma_3) \vdash_{\max(n_1,n_2,n_3)}\ \mathrm{case}\ (t, x.t_1, y.t_2) : \tau}\ \textsc{case} \qquad \frac{\vdash \tau\ \mathtt{wf}}{\Gamma \vdash_0 \mathrm{nil} : \tau\,\mathrm{list}}\ \textsc{nil}$$

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau\,\mathrm{list}}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathrm{cons}\,(t_1, t_2) : \tau\,\mathrm{list}}\ \textsc{cons}$$

$\boxed{\tau \subseteq \tau}$

$$\frac{}{(!_i \tau_1, !_j \tau_2) \subseteq !_{\max(i,j)} (\tau_1, \tau_2)}\ \textsc{S-pair} \qquad \frac{}{!_i \tau_1\,\mathrm{list} \subseteq !_i \tau_1\,\mathrm{list}}\ \textsc{S-list}$$

Figure 6: New added components

$$\boxed{\Gamma \vdash_{n,m} t : \tau}$$

$$\boxed{\Gamma ::= \emptyset \mid \Gamma, x : \tau \mid \Gamma, x : [\tau]_p}$$

$$\frac{}{\Gamma \vdash_{n,m} c : b}\ \textsc{const} \qquad \frac{\Gamma, x : \tau_1 \vdash_n t : \tau_2}{\Gamma \vdash_{n,m} \lambda x.t : \tau_1 \multimap \tau_2}\ \textsc{abs} \qquad \frac{[\Gamma] \vdash_n t : \tau}{\Delta, p + [\Gamma] \vdash_{n+p} !t : !_p \tau}\ \textsc{pr}$$

$$\frac{}{\Gamma, x : \tau \vdash_n x : \tau}\ \textsc{var} \qquad \frac{[\Gamma] \vdash_n t : query}{\Delta, 1 + [\Gamma] \vdash_{n+1} M(t) : b}\ \mathrm{MT} \qquad \frac{}{\Gamma \vdash_n q : query}\ \textsc{query}$$

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau_1 \multimap \tau_2 \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau_1}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} t_1\ t_2 : \tau_2}\ \textsc{app} \qquad \boxed{\frac{\Gamma, x : \tau \vdash_n t : \tau}{\Gamma, x : [\tau]_0 \vdash_n t : \tau}\ \textsc{der}}$$

$$\frac{\Gamma_1 \vdash_{n_1} t : !_p \tau \qquad \Gamma_2, x : [\tau]_p \vdash_{n_2} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !x = t\ \mathsf{in}\ t' : \tau'}\ \textsc{let-b}$$

$$\frac{\Gamma_1 \vdash_{n_1} t : \tau_1 \times \tau_2 \qquad \Gamma_2, x_1 : \tau_1, x_2 : \tau_2 \vdash_{n_2} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'}\ \textsc{let-p}$$

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau_1 \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau_2}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} (t_1, t_2) : \tau_1 \times \tau_2}\ \textsc{pair}$$

$$\frac{\Gamma_1 \vdash_{n_1} t : b \qquad \Gamma_2 \vdash_{n_2} t_i : b}{\max(n_2 + \Gamma_1, \Gamma_2) \vdash_{(n_1+n_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b}\ \textsc{case-const}$$

$$\frac{\Gamma_1 \vdash_{n_1} t : b \qquad \Gamma_2 \vdash_{n_2} t_i : query}{\max(\Gamma_1, \Gamma_2) \vdash_{(n_1+n_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : query}\ \textsc{case-query}$$

$$\frac{i :: \mathbb{N}; \Gamma \vdash_n t : \tau \qquad i \notin \mathrm{FIV}(\Gamma)}{\Gamma \vdash_n \Lambda.t : \forall i{::}\mathbb{N}.\tau}\ \textsc{iabs} \qquad \frac{\Gamma \vdash_n t : \forall i{::}\mathbb{N}.\tau \qquad \vdash I :: \mathbb{N}}{\Gamma \vdash_n t\,[\,] : \tau\{I/i\}}\ \textsc{iapp}$$

$$\frac{\Gamma \vdash_n t : \tau \qquad \Gamma' \sqsubseteq \Gamma \qquad \vDash n \leq n' \qquad \tau \sqsubseteq \tau'}{\Gamma' \vdash_{n'} t : \tau'}\ \textsc{sub}$$

Figure 7: Typing judgment

$$\frac{}{\tau <: \tau}\ \text{S-ID} \qquad \frac{A <: B \qquad q \le p}{!_p A <: !_q B}\ \text{S-B} \qquad \frac{A' <: A \qquad B <: B'}{A \multimap B <: A' \multimap B'}\ \text{S-ARROW}$$

$$\frac{A \subseteq B \qquad q \le p}{[A]_p \subseteq [B]_q}\ \text{S-D} \qquad \frac{}{\Gamma \subseteq \Gamma}\ \text{S-IDC} \qquad \frac{}{\Gamma \subseteq \emptyset}\ \text{S-EMPTY} \qquad \frac{A \subseteq B \qquad \Gamma \subseteq \Delta}{\Gamma, x : A \subseteq \Delta, x : B}\ \text{S-CTX}$$

$$\frac{\Delta \subseteq \Gamma}{x : \tau, \Delta \subseteq \Gamma}\ \text{S-XCTX1} \qquad \frac{\Delta \subseteq \Gamma}{x : [\tau]_p, \Delta \subseteq \Gamma}\ \text{S-XCTX2}$$

Figure 8: sub typing

$$\boxed{t \Downarrow^m v}$$

$$\frac{}{c \Downarrow^0 c}\ \textsc{E-const} \qquad \frac{}{q \Downarrow^0 q}\ \textsc{E-query} \qquad \frac{}{\lambda x.t \Downarrow^0 \lambda x.t}\ \text{E-ABS}$$

$$\frac{}{!t \Downarrow^0 !t}\ \textsc{E-bang} \qquad \frac{t_1 \Downarrow^{m_1} v_1 \qquad t_2 \Downarrow^{m_2} v_2}{(t_1, t_2) \Downarrow^{\max(m_1,m_2)} (v_1, v_2)}\ \textsc{E-pair}$$

$$\frac{t_1 \Downarrow^{m_1} \lambda x.t \qquad t_2 \Downarrow^{m_2} v \qquad t[v/x] \Downarrow^{m_3} v'}{t_1\ t_2 \Downarrow^{\max(m_1,m_2)+m_3} v'}\ \textsc{E-app}$$

$$\boxed{\frac{t_1 \Downarrow^{m_1} !t_3 \qquad t_3 \Downarrow^{m_2} v' \qquad t_2[v'/x] \Downarrow^{m_3} v}{\mathsf{let}\ !x = t_1\ \mathsf{in}\ t_2 \Downarrow^{\max(m_1+m_2,m_3)} v}\ \textsc{E-let-bang}}$$

$$\frac{t \Downarrow^{m_1} (v_1, v_2) \qquad t'[v_1/x_1][v_2/x_2] \Downarrow^{m2} v}{\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' \Downarrow^{\max(m_1,m_2)} v}\ \textsc{E-let-p}$$

$$\frac{t \Downarrow^m v \qquad t_i \Downarrow^{m_i} v_i}{\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} \Downarrow^{m+\max(m_i)} \mathsf{case}\ v\ \mathsf{of}\ \{c_i \Rightarrow v_i\}_{c_i \in b}}\ \textsc{E-case} \qquad \frac{}{\mathsf{fix} f(x).t \Downarrow^0 \mathsf{fix} f(x).t}\ \textsc{E-fix}$$

$$\frac{}{x \Downarrow^0 x}\ \textsc{E-x} \qquad \frac{}{\Lambda.t \Downarrow^0 \Lambda.t}\ \text{E-ILAM} \qquad \frac{t \Downarrow^m \Lambda.t'}{t[] \Downarrow^m t'}\ \textsc{E-iapp} \qquad \frac{t \Downarrow^m v \qquad M(v) \Downarrow^1 v'}{M(t) \Downarrow^{m+1} v'}\ \textsc{E-mech}$$

Figure 9: Evaluation Rules

$$\frac{b \Downarrow^0 true \qquad t_1 \Downarrow^m v_1}{\text{if } b \text{ then } t_1 \text{ else } t_2 \Downarrow^m v_1}\ \textsc{E-if-true} \qquad \frac{b \Downarrow^0 false \qquad t_2 \Downarrow^m v_2}{\text{if } b \text{ then } t_1 \text{ else } t_2 \Downarrow^m v_2}\ \textsc{E-if-false} \qquad \frac{}{\text{nil} \Downarrow^0 \text{nil}}\ \textsc{E-nil}$$

$$\frac{t_1 \Downarrow^{m_1} v_1 \qquad t_2 \Downarrow^{m_2} v_2}{\text{cons}(t_1, t_2) \Downarrow^{\max(m_1,m_2)} \text{cons}(v_1, v_2)}\ \textsc{E-cons} \qquad \frac{t_2 \Downarrow^{m_2} v_2 \qquad t[v_2/x] \Downarrow^m v}{\mathsf{let}\ x = t_2\ \mathsf{in}\ t \Downarrow^{\max(m_2,m)} v}\ \textsc{E-let}$$

Figure 10: New Added Evaluation Rules

$$
\begin{aligned}
\llbracket \tau \rrbracket_{\epsilon} &= \{ e \mid \exists v. e \Downarrow v \wedge v \in \llbracket \tau \rrbracket_{v} \} \\
\llbracket b \rrbracket_{v} &= \{ v \mid v = T_{b} \} \\
\llbracket query \rrbracket_{v} &= \{ v \mid v = T_{query} \} \\
\llbracket \tau_1 \to \tau_2 \rrbracket_{v} &= \{ \lambda x.t \mid \forall v \in \llbracket \tau \rrbracket_{v}. t[v/x] \in \llbracket \tau_2 \rrbracket_{\epsilon} \} \\
\llbracket\, !_{n} \tau \rrbracket_{v} &= \{ !t \mid t \in \llbracket \tau \rrbracket_{\epsilon} \} \\
\llbracket \forall i{::}\mathbb{N}. \tau \rrbracket_{v} &= \{ \Lambda.t \mid \forall I. \vdash i :: \mathbb{N}. t[I/i] \in \llbracket \tau \rrbracket_{\epsilon} \} \\
\llbracket \tau_1 * \tau_2 \rrbracket_{v} &= \{ (v_1, v_2) \mid v_1 \in \llbracket \tau_1 \rrbracket_{v} \wedge v_2 \in \llbracket \tau_2 \rrbracket_{v} \} \\
\llbracket \cdot \rrbracket &= \{ \emptyset \} \\
\llbracket \Gamma, x : [\tau]_{p} \rrbracket &= \{ \gamma[x \to v] \mid v \in \llbracket \tau \rrbracket_{v} \wedge \gamma \in \llbracket \Gamma \rrbracket \} \\
\boxed{\llbracket \Gamma, x : [\tau]_{p} \rrbracket} &= \{ \gamma[x \to v] \mid v \in \llbracket !_{p} \tau \rrbracket_{v} \wedge \gamma \in \llbracket \Gamma \rrbracket \} \\
\llbracket \Gamma, x : \tau \rrbracket &= \{ \gamma[x \to v] \mid v \in \llbracket \tau \rrbracket_{v} \wedge \gamma \in \llbracket \Gamma \rrbracket \} \\
\gamma \vDash \Gamma &\triangleq dom(\gamma) = dom(\Gamma) \wedge \forall x \in dom(\Gamma). \gamma(x) \in \llbracket \Gamma(x) \rrbracket_{v}
\end{aligned}
$$

Figure 11: denotations

Lemma 1.

1. *If* $\vdash_{n,m} v : b$ *then* $\exists T_b : v = T_b$.

2. *If* $\vdash_{n,m} v : query$ *then* $\exists T_{query} : v = T_{query}$

Lemma 2 (Depth Definition). *If* $\Gamma \vdash_{n,m} t : \tau$ *then* $\mathsf{depth}(t) \leq n$

Proof. It is proved by the induction on the structure of the typing derivation.
Case

$$\frac{\Gamma \vdash_n t : \tau \;\; (\star)}{p + \Gamma \vdash_n !t : !_p \tau}\,\text{PR}$$

TS: $\mathsf{depth}(!t) \leq n$.
By IH on $(\star)$, we get $\mathsf{depth}(t) \leq n$
This case is proved because $\mathsf{depth}(!t) = \mathsf{depth}(t)$.
Case

$$\frac{}{\Gamma \vdash_0 c : b}\,\text{CONST}$$

TS: $\mathsf{depth}(c) \leq 0$
It is already proved by the definition of $\mathsf{depth}(c)$.
Case

$$\frac{\Gamma, x : [\tau_1]_0 \vdash_n t : \tau_2 \;\; (\star)}{\Gamma \vdash_n \lambda x.t : \tau_1 \multimap \tau_2}\,\text{ABS}$$

TS: $\mathsf{depth}(\lambda x.t) \leq n$.
By IH on $(\star)$ instantiating the context with $\Gamma, x : [\tau]_0$, we get : $\mathsf{depth}(t) \leq n$.
This case is proved by the definition of $\mathsf{depth}(\lambda x.t)$.
Case

$$\frac{\Gamma \vdash_n t : query \;\; (\star)}{1 + \Gamma \vdash_{n+1} M(t) : b}\,\text{MT}$$

TS: $\mathsf{depth}(M(t)) \leq n+1$.
By IH on $(\star)$, we get $\mathsf{depth}(t) \leq n$.
It is proved by the definition of $\mathsf{depth}(M(t)) = \mathsf{depth}(t) + 1$.
Case

$$\frac{}{\Gamma, x : [\tau]_p \vdash_0 x : \tau}\,\text{VAR}$$

TS: $\mathsf{depth}(x) \leq 0$.
It is proved by the definition of $\mathsf{depth}(x)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau_1 \to \tau_2 \;\; (\star) \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau_1 \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} t_1\, t_2 : \tau_2}\,\text{APP}$$

TS: $\mathsf{depth}(t_1\, t_2) \leq \max(n_1, n_2)$.
By IH on $(\star)$ and $(\diamond)$, we get: $\mathsf{depth}(t_1) \leq n_1$ $(\star\star)$ and $\mathsf{depth}(t_2) \leq n_2 (\diamond\diamond)$.
Unfold the definition of $\mathsf{depth}(t_1\, t_2) = \max(\mathsf{depth}(t_1), \mathsf{depth}(t_2))$.
This case is proved by the $(\star\star)$ and $(\diamond\diamond)$.
Case

$$\frac{}{\Gamma \vdash_0 q : query}\,\text{QUERY}$$

TS: $\mathsf{depth}(q) \leq 0$.
This is proved by the definition of $\mathsf{depth}(q)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : !_p\tau \;\; (\star) \qquad \Gamma_2, x : [\tau]_p \vdash_{n_2} t' : \tau' \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !x = t\ \mathsf{in}\ t' : \tau'}\ \textsc{let}$$

To show: $\mathsf{depth}(\mathsf{let}\ !x = t\ \mathsf{in}\ t') \leq \max(n_1, n_2)$.
By induction Hypothesis on $(\star)$ and $(\diamond)$, we get: $\mathsf{depth}(t) \leq n_1$ $(\star\star)$ and $\mathsf{depth}(t') \leq n_2$ $(\diamond\diamond)$.
Unfolding the definition of $\mathsf{depth}(\mathsf{let}\ !x = t\ \mathsf{in}\ t') = \max(\mathsf{depth}(t), \mathsf{depth}(t'))$.
This case is proved by the $(\star\star)$ and $(\diamond\diamond)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : \tau_1 \times \tau_2 \;\; (\star) \qquad \Gamma_2, x_1 : \tau_1, x_2 : \tau_2 \vdash_{n_2} t' : \tau' \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'}\ \textsc{let-p}$$

To show: $\mathsf{depth}(\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t') \leq \max(n_1, n_2)$.
By induction Hypothesis on $(\star)$ and $(\diamond)$, we get: $\mathsf{depth}(t) \leq n_1$ $(\star\star)$ and $\mathsf{depth}(t') \leq n_2$ $(\diamond\diamond)$.
Unfolding the definition of $\mathsf{depth}(\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t') = \max(\mathsf{depth}(t), \mathsf{depth}(t'))$.
This case is proved by the $(\star\star)$ and $(\diamond\diamond)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau_1 \;\; (\star) \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau_2 \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} (t_1, t_2) : \tau_1 \times \tau_2}\ \textsc{pair}$$

To show: $\mathsf{depth}((t_1, t_2)) \leq \max(n_1, n_2)$.
By induction Hypothesis on $(\star)$ and $(\diamond)$, we get: $\mathsf{depth}(t_1) \leq n_1$ $(\star\star)$ and $\mathsf{depth}(t_2) \leq n_2$ $(\diamond\diamond)$.
Unfolding the definition of $\mathsf{depth}((t_1, t_2)) = \max(\mathsf{depth}(t_1), \mathsf{depth}(t_2))$.
This case is proved by the $(\star\star)$ and $(\diamond\diamond)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : b \;\; (\star) \qquad \Gamma_2 \vdash_{n_2} t_i : b \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b}\ \textsc{case-const}$$

To show: $\mathsf{depth}(\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b}) \leq \max(n_1, n_2)$.
By induction Hypothesis on $(\star)$ and $(\diamond)$, we get: $\mathsf{depth}(t) \leq n_1$ $(\star\star)$ and $\mathsf{depth}(t_i) \leq n_2$ $(\diamond\diamond)$.
Unfolding the definition of $\mathsf{depth}(\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b}) = \max(\mathsf{depth}(t), \mathsf{depth}(t_i))$.
This case is proved by the $(\star\star)$ and $(\diamond\diamond)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : b \;\; (\star) \qquad \Gamma_2 \vdash_{n_2} t_i : query \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : query}\ \textsc{case-query}$$

To show: $\mathsf{depth}(\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b}) \leq \max(n_1, n_2)$.
By induction Hypothesis on $(\star)$ and $(\diamond)$, we get: $\mathsf{depth}(t) \leq n_1$ $(\star\star)$ and $\mathsf{depth}(t_i) \leq n_2$ $(\diamond\diamond)$.
Unfolding the definition of $\mathsf{depth}(\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b}) = \max(\mathsf{depth}(t), \mathsf{depth}(t_i))$.
This case is proved by the $(\star\star)$ and $(\diamond\diamond)$.

Case

$$\frac{i :: \mathbb{N}; \Gamma \vdash_n t : \tau \;\; (\diamond) \qquad i \notin FV(\Gamma)}{\Gamma \vdash_n \Lambda.t : \forall i{::}\mathbb{N}.\,\tau}\ \textsc{iabs}$$

To show: $\mathsf{depth}(\Lambda.t) \leq n$.
By induction Hypothesis on $(\diamond)$, we get: $\mathsf{depth}(t) \leq n$ $(\diamond\diamond)$.
Unfolding the definition of $\mathsf{depth}(\Lambda.t) = \mathsf{depth}(t)$.
This case is proved by the $(\diamond\diamond)$.

Case

$$\frac{\Gamma \vdash_n t : \forall i{::}\mathbb{N}.\,\tau \;\; (\diamond) \qquad \vdash I :: \mathbb{N}}{\Gamma \vdash_n t\,[\,] : \tau\{I/i\}}\ \textsc{iapp}$$

To show: $\mathsf{depth}(t\,[\,]) \leq n$.
By induction Hypothesis on $(\diamond)$, we get: $\mathsf{depth}(t) \leq n$ $(\diamond\diamond)$.
Unfolding the definition of $\mathsf{depth}(t\,[\,]) = \mathsf{depth}(t)$.
This case is proved by the $(\diamond\diamond)$.

$\square$

Lemma 3 (Depth Weakening1). $\Gamma \vdash_{n_1,m} t : \tau \wedge n_1 \leq n_2 \implies \Gamma \vdash_{n_2,m} t : \tau$

Proof. By induction on $\Gamma \vdash_{n_1} t : \tau$. □

Lemma 4 (Depth Weakening2). $\Gamma, x : [\tau]_{p_1} \vdash_n t : \tau \wedge p_1 \leq p_2 \implies \exists m. n \leq m \; s.t. \; \Gamma, x : [\tau]_{p_2} \vdash_n t : \tau$

Proof. By induction on $\Gamma, x : [\tau]_{p_1} \vdash_n t : \tau$.
Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : !_p \tau_1 \qquad \Gamma_2, y : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t_1 \ \mathsf{in}\ t_2 : \tau_2} \text{ LET-B}$$

where $\tau = \tau_2$, $t =$let $!y = t_1$ in t_2.

Subcase 1: x $\not\in$ dom(Δ_2)

$$\frac{\Gamma_1', x : [\tau]_{p1} \vdash_{n_1} t_1 : !_p \tau_1 \;\; (\star) \qquad \Gamma_2, y : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2 \;\; (\diamond)}{\max(\Gamma_1', x : [\tau]_{p1}, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t_1 \ \mathsf{in}\ t_2 : \tau_2} \text{ LET-B}$$

To show:$\max(\Gamma_1', x : [\tau]_{p2}, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t_1 \ \mathsf{in}\ t_2 : \tau_2$
By ih on $(\star)$, we get: $\Gamma_1', x : [\tau]_{p2} \vdash_{n_1} t_1 : !_p \tau_1 \;\; (\star\star)$.
Applying the rule LET-B on $(\star\star)$ and $(\diamond)$, this case is proved.

Subcase 2: x $\not\in$ dom(Δ_1)

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : !_p \tau_1 \;\; (\star) \qquad \Gamma_2', x : [\tau]_{p1}, y : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2 \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2', x : [\tau]_{p1}) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t_1 \ \mathsf{in}\ t_2 : \tau_2} \text{ LET-B}$$

To show:$\max(\Gamma_1', x : [\tau]_{p2}, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t_1 \ \mathsf{in}\ t_2 : \tau_2$
By ih on $(\diamond)$, we get: $\Gamma_2', x : [\tau]_{p2}, y : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2 \;\; (\diamond\diamond)$.
Applying the rule LET-B on $(\star)$ and $(\diamond\diamond)$, this case is proved.

Subcase 3

$$\frac{\Gamma_1', x : [\tau]_{p1} \vdash_{n_1} t_1 : !_p \tau_1 \qquad \Gamma_2, x : [\tau]_{p1}, y : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2}{\max(\Gamma_1', \Gamma_2, x : [\tau]_{p1}) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t_1 \ \mathsf{in}\ t_2 : \tau_2} \text{ LET-B}$$

To show:$\max(\Gamma_1', x : [\tau]_{p2}, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t_1 \ \mathsf{in}\ t_2 : \tau_2$
By ih on $(\diamond)$, we get: $\Gamma_2', x : [\tau]_{p2}, y : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2 \;\; (\diamond\diamond)$.
Applying the rule LET-B on $(\star)$ and $(\diamond\diamond)$, this case is proved.

Case

$$\frac{[\Gamma] \vdash_n t : \tau}{\Delta, p + [\Gamma] \vdash_{n+p} !t : !_p \tau} \text{ PR}$$

□

Lemma 5 (Context weakening - 1). $\Gamma \vdash_{n,m} t : \tau \implies \Gamma, x : \tau \vdash_{n,m} t : \tau$

Proof. By induction on $\Gamma \vdash_n t : \tau$ and using ih. □

Lemma 6 (Context weakening - 2). $\Gamma \vdash_{n,m} t : \tau \implies \Gamma, x : [\tau]_p \vdash_{n,m} t : \tau$

Proof. By induction on $\Gamma \vdash_n t : \tau$, using ih. □

Lemma 7 (Context exchange). $\Gamma, x : \tau_1, \Delta, y : \tau_2 \vdash_{n,m} t : \tau \implies \Gamma, y : \tau_2, \Delta, x : \tau_1 \vdash_{n,m} t : \tau$

Lemma 8. *If* $\Gamma \vdash_n t : \tau$ *and* $\gamma \vDash \Gamma$, *then* $\cdot \vdash_n \gamma(t) : \tau$

Lemma 9. *If* $\Gamma \subseteq \Gamma'$*, and* $\Gamma' \vdash_n t : \tau$*, then* $\exists m.n \leq m$ *s.t.* $\Gamma \vdash_m t : \tau$*.*

Proof. Induction on $\Gamma \subseteq \Gamma'$.
Case

$$\frac{}{\Gamma \subseteq \Gamma} \text{ S-IDC}$$

$\Gamma' = \Gamma$.
This case is proved.

Case

$$\frac{\Delta \subseteq \Gamma}{x : \tau', \Delta \subseteq \Gamma} \text{ S-XCTX1}$$

We have $\Gamma \vdash_n t : \tau$.
TS: $x : \tau', \Delta \vdash_n t : \tau$.
By ih, we get : $\exists m.m \geq n$. s.t. $\Delta \vdash_m t : \tau$, this case is proved by Lemma 5.
Case

$$\frac{\Delta \subseteq \Gamma}{x : [\tau']_p, \Delta \subseteq \Gamma} \text{ S-XCTX2}$$

We have $\Gamma \vdash_n t : \tau$.
TS: $x : \tau', \Delta \vdash_n t : \tau$.
By ih, we get : $\exists m.m \geq n$. s.t. $\Delta \vdash_m t : \tau$, this case is proved by Lemma 6.
Case

$$\frac{}{\Gamma \subseteq \emptyset} \text{ S-EMPTY}$$

It is proved by Lemma 5 and Lemma 6 several times for every variables in Γ.

Case

$$\frac{A \subseteq B \qquad \Gamma \subseteq \Delta}{\Gamma, x : A \subseteq \Delta, x : B} \text{ S-CTX}$$

We have: $\Delta, x : B \vdash_n t : \tau$.
TS: $\exists m.m \geq n$ s.t. $\Gamma, x : A \vdash_m t : \tau$
Induction on $A \subseteq B$.

SubCase

$$\frac{}{\tau \subseteq \tau} \text{ S-ID}$$

SubCase

$$\frac{A \subseteq B \qquad q \leq p}{!_p A \subseteq !_q B} \text{ S-B}$$

SubCase

$$\frac{A' \subseteq A \qquad B \subseteq B'}{A \multimap B \subseteq A' \multimap B'} \text{ S-ARROW}$$

SubCase

$$\frac{A \subseteq B \qquad q \leq p}{[A]_p \subseteq [B]_q} \text{ S-D}$$

□

Theorem 0.1 (Type Safety)**.** *If* $\cdot \vdash_{n,m} t : \tau$ *then* $\exists F. t \Downarrow F \wedge \vdash_{n,m} F : \tau$

Proof. We prove this theorem by prove Normalization and Preservation. □

Corollary 0.1.1. *If* $\cdot \vdash_{n,m} t : b$ *then* $\exists T_b . t \Downarrow T_b \wedge \mathsf{depth}(T_b) \leq n$

Theorem 0.2 (Normalization)**.** *If* $\cdot \vdash_{n,m} t : \tau$ *then* $\exists F : t \Downarrow F$

We prove two theorems instead.

Theorem 0.3. *If* $\gamma(t) \in [\![\tau]\!]_\epsilon$*, then* $\exists F.\, \gamma(t) \Downarrow F$*.*

Proof. It is proved by unfolding the definition of $[\![\tau]\!]_\epsilon$. □

Theorem 0.4. *If* $\Gamma \vdash_n t : \tau$ *and* $\gamma \vDash \Gamma$*, then* $\gamma(t) \in [\![\tau]\!]_\epsilon$*.*

Proof. Proof by induction on $\Gamma \vdash_n t : \tau$
We have $\gamma \vDash \Gamma$ (♠).
Case

$$\frac{\Gamma, x : \tau_1 \vdash_n t_1 : \tau_2 \;\; (\star)}{\Gamma \vdash_{n,m} \lambda x.t_1 : \tau_1 \multimap \tau_2} \textsc{abs}$$

TS : $\gamma(\lambda x.t) \in [\![\tau_1 \multimap \tau_2]\!]_\epsilon$
Because $\gamma(\lambda x.t_1) = \lambda x.\gamma(t_1)$ is value, unfold the definition of $[\![\tau_1 \multimap \tau_2]\!]_\epsilon$, STS: $\gamma(\lambda x.t_1) \in [\![\tau_1 \multimap \tau_2]\!]_v$
Unfold the definition of $[\![\tau_1 \multimap \tau_2]\!]_v$, STS: $\forall v.v \in [\![\tau_1]\!]_v.\gamma(t_1)[v/x] \in [\![\tau_2]\!]_\epsilon$.
Pick v s.t $v \in [\![\tau_1]\!]_v$. STS: $\gamma(t_1)[v/x] \in [\![\tau_2]\!]_\epsilon$.
We have $\gamma[x \to v] \vDash \Gamma, x : \tau_1$ $(\diamond)$ because $\gamma \vDash \Gamma$ and $v \in [\![\tau_1]\!]_v$(the assumption).
By IH on $(\star)$ and $(\diamond)$, we have :

$$\gamma[v/x](t_1) \in [\![\tau_2]\!]_\epsilon$$

Because $\gamma[v/x](t_1) = \gamma(t_1[v/x])$, this case is proved.

Case

$$\frac{[\Gamma] \vdash_n t_1 : \tau_1}{\Delta, p + [\Gamma] \vdash_{n+p} !t_1 : !_p\tau_1} \textsc{pr}$$

We assume $\gamma \vDash \Delta, p + [\Gamma]$ (♠).
TS : $\gamma(!t_1) \in [\![!_p\tau_1]\!]_\epsilon$
Because $\gamma(!t_1) = !\gamma(t_1)$ is value, unfold the definition of $[\![!_p\tau_1]\!]_\epsilon$, STS: $!\gamma(t_1) \in [\![!_p\tau_1]\!]_v$
Unfold the definition of $[\![!_p\tau_1]\!]_v$, STS: $\gamma(t_1) \in [\![\tau_1]\!]_\epsilon$.
We extend the context of the premise to $\Delta, p + [\Gamma]$ by Lemma 5 and Lemma 6. By ih, we get : $\gamma(t_1) \in [\![\tau_1]\!]_\epsilon$ $(\star)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : !_p\tau_1 \;\; (\star) \qquad \Gamma_2, x : [\tau_1]_p \vdash_{n_2} t_2 : \tau' \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !x = t_1\ \mathsf{in}\ t_2 : \tau'} \textsc{let}$$

We assume $\gamma \vDash \max(\Gamma_1, \Gamma_2)$ (♠). TS: $\gamma(\mathsf{let}\ !x = t_1\ \mathsf{in}\ t_2) \in [\![\tau']\!]_\epsilon$.
Unfold $[\![\tau']\!]_\epsilon$, STS: $\exists F.\, \gamma(\mathsf{let}\ !x = t_1\ \mathsf{in}\ t_2) \Downarrow F \wedge F \in [\![\tau']\!]_v$.

Extend the context of $(\star)$ to $\max(\Gamma_1, \Gamma_2)$ using Lemma 5 and Lemma 6 or the rule (SUB). By ih on $(\star)$ we get : $\gamma(t_1) \in [\![!_p\tau_1]\!]_\epsilon$ (1)
Unfold (1), we get: $\exists F.\, \gamma(t_1) \Downarrow F \wedge F \in [\![!_p\tau_1]\!]_v$ (2).
Unfold (2), we know: $\exists !t_3.\, \gamma(t_1) \Downarrow !t_3 \wedge t_3 \in [\![\tau_1]\!]_\epsilon$ (3).
Unfold (3), we know: $\exists !t_3.\, \gamma(t_1) \Downarrow !t_3 \wedge \exists F'.\, t_3 \Downarrow F' \wedge F' \in [\![\tau_1]\!]_v$ (4).
Pick F' in (4). Extend the context of $(\diamond)$ to $\max(\Gamma_1, \Gamma_2), x : [\tau_1]_p$ s.t. $\gamma[F'/x] \vDash \max(\Gamma_1, \Gamma_2), x : [\tau_1]_p$. By ih on $(\diamond)$ we get : $\gamma[F'/x](t_2) \in [\![\tau']\!]_\epsilon$ (5).
Unfold (5), we know $\exists F.\, \gamma[F'/x](t_1') \Downarrow F \wedge F \in [\![\tau']\!]_v$ (6).

This case is proved using the following evaluation rule E-LET-BANG and (6).

$$\frac{\gamma(t_1) \Downarrow^{m_1} !t_3 \qquad t_3 \Downarrow^{m_2} F' \qquad \gamma(t_2[F'/x]) \Downarrow^{m_3} F}{\mathsf{let}\ !x = \gamma(t_1)\ \mathsf{in}\ \gamma(t_2) \Downarrow^{m_1+m_2+m_3} F} \textsc{E-let-bang}$$

Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : !_p \tau_1 \;\; (\star) \qquad \Gamma_2, x : [\tau_1]_p \vdash_{n_2} t_1' : \tau' \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\; !x = t_1 \;\mathsf{in}\; t_1' : \tau'} \;\textsc{let}$$

We assume $\gamma \vDash \max(\Gamma_1, \Gamma_2)$ ($\spadesuit$). TS: $\gamma(\mathsf{let}\; !x = t_1 \;\mathsf{in}\; t_1') \in [\![\tau']\!]_\epsilon$.
Unfold $[\![\tau']\!]_\epsilon$, STS: $\exists F.\, \gamma(\mathsf{let}\; !x = t \;\mathsf{in}\; t') \Downarrow F \wedge F \in [\![\tau']\!]_v$.

Pick γ_1 s.t. $\gamma_1 \vDash \Gamma_1$. By ih on ($\star$) we get : $\gamma_1(t_1) \in [\![!_p \tau_1]\!]_\epsilon$.
By Lemma 9 and $\Gamma_1 \subseteq \max(\Gamma_1, \Gamma_2)$, we know: $\gamma(t_1) \in [\![!_p \tau_1]\!]_\epsilon$ (1)
Unfold (1), we get: $\exists F.\, \gamma(t_1) \Downarrow F \wedge F \in [\![!_p \tau_1]\!]_v$ (2).

Pick γ_2 s.t. $\gamma_2 \vDash \Gamma_2 \implies \gamma_2[F/x] \vDash \Gamma_2, x : [\tau_1]_p$. By ih on ($\diamond$) we get : $\gamma_2[F/x](t_1') \in [\![\tau']\!]_\epsilon$ (3).
By Lemma 9 and $\Gamma_2, x : [\tau_1]_p \subseteq \max(\Gamma_1, \Gamma_2), x : [\tau_1]_p$, we know: $\gamma[F/x](t_1') \in [\![\tau']\!]_\epsilon$ (4)

It is proved by unfolding (2),(4) and the following evaluation rule E-LET-BANG.

$$\frac{\gamma(t_1) \Downarrow^{m_1} !t_2 \qquad \gamma[!t_2/x](t_1') \Downarrow^{m_3} F'}{\mathsf{let}\; !x = \gamma(t_1) \;\mathsf{in}\; \gamma(t_1') \Downarrow^{m_1+m_2+m_3} F'} \;\textsc{E-let-bang}$$

Case

$$\frac{[\Gamma] \vdash_n t : query}{\Delta, 1 + [\Gamma] \vdash_{n+1} M(t) : b} \;\textsc{MT}$$

Assume $\gamma \vDash \Delta, 1 + [\Gamma]$ ($\spadesuit$).
TS: $\gamma(M(t)) \in [\![b]\!]_\epsilon$.
STS: $\exists F.\, M(\gamma(t)) \Downarrow F \wedge F \in [\![b]\!]_v \implies \exists T_b.\, M(\gamma(t)) \Downarrow T_b$.
We assume $\gamma' \vDash [\Gamma]$, by ih we get : $\gamma'(t) \in [\![query]\!]_\epsilon$ (1).
By Lemma 9 and $[\Gamma] \subseteq \Delta, 1 + [\Gamma]$, we get $\gamma(t) \in [\![query]\!]_\epsilon$ (2)
Unfold (2), we know: $\exists F.\, \gamma(t) \Downarrow F \wedge F \in [\![query]\!]_v \implies F = T_{query}$ (3).

It is proved by E-MECH and (3), $T_b = M(T_{query})$.

$$\frac{\gamma(t) \Downarrow^m T_{query}}{M(t) \Downarrow^m M(T_{query})} \;\textsc{E-mech}$$

Case

$$\frac{}{\Gamma, x : \tau \vdash_n x : \tau} \;\textsc{var}$$

Assume $\gamma \vDash \Gamma, x : \tau$.
TS: $\gamma(x) \in [\![\tau]\!]_\epsilon$.
Unfold the definition of $\gamma \vDash \Gamma, x : \tau$, we know : $\gamma(x) \in [\![\tau]\!]_v$.
STS: $F \in [\![\tau]\!]_\epsilon$.

It is proved by the assumption and the evaluation rule E-Value.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : b \;\; (\star) \qquad \Gamma_2 \vdash_{n_2} t_i : b \;\; (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{case}\; t \;\mathsf{of}\; \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \;\textsc{case-const}$$

We assume $\gamma \vDash \max(\Gamma_1, \Gamma_2)$.
TS: $\gamma(\mathsf{case}\; t \;\mathsf{of}\; \{c_i \Rightarrow t_i\}_{c_i \in b}) \in [\![b]\!]_\epsilon \implies \mathsf{case}\; \gamma(t) \;\mathsf{of}\; \{c_i \Rightarrow \gamma(t_i)\}_{c_i \in b} \in [\![b]\!]_\epsilon$.
We assume $\gamma_1 \vDash \Gamma_1$ and $\gamma_2 \vDash \Gamma_2$.
By ih on ($\star$), we get : $\gamma_1(t) \in [\![b]\!]_\epsilon$.
By ih on ($\diamond$), we get : $\gamma_2(t_i) \in [\![b]\!]_\epsilon$.
By Lemma 9 and $\Gamma_1 \subseteq \max(\Gamma_1, \Gamma_2)$ and $\Gamma_2 \subseteq \max(\Gamma_1, \Gamma_2)$, we get: $\gamma(t) \in [\![b]\!]_\epsilon$ (1) and $\gamma(t_i) \in [\![b]\!]_\epsilon$ (2).
It is proved by unfolding (1) and (2) and using the evaluation rule E-CASE and the definition of T_b.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : b\,(\star) \qquad \Gamma_2 \vdash_{n_2} t_i : query\ \ (\diamond)}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : query}\ \textsc{case-query}$$

We assume $\gamma \vDash \max(\Gamma_1, \Gamma_2)$.
TS: $\gamma(\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b}) \in [\![query]\!]_\epsilon \implies \mathsf{case}\ \gamma(t)\ \mathsf{of}\ \{c_i \Rightarrow \gamma(t_i)\}_{c_i \in b} \in [\![query]\!]_\epsilon$.
We assume $\gamma_1 \vDash \Gamma_1$ and $\gamma_2 \vDash \Gamma_2$.
By ih on $(\star)$, we get : $\gamma_1(t) \in [\![b]\!]_\epsilon$.
By ih on $(\diamond)$, we get : $\gamma_2(t_i) \in [\![query]\!]_\epsilon$.
By Lemma 9 and $\Gamma_1 \subseteq \max(\Gamma_1, \Gamma_2)$ and $\Gamma_2 \subseteq \max(\Gamma_1, \Gamma_2)$, we get: $\gamma(t) \in [\![b]\!]_\epsilon$ (1) and $\gamma(t_i) \in [\![query]\!]_\epsilon$ (2).
It is proved by unfolding (1) and (2) and using the evaluation rule E-CASE and the definition of T_{query}.

Case

$$\frac{\Gamma \vdash_n t : \tau\ \ (\star) \qquad \Gamma' \subseteq \Gamma \qquad \vDash n \le n' \qquad \tau \subseteq \tau'}{\Gamma' \vdash_{n'} t : \tau'}\ \textsc{sub}$$

We assume $\gamma \vDash \Gamma'$.
To show $\gamma(t) \in [\![\tau']\!]_\epsilon$.
We assume $\gamma' \vDash \Gamma$. By ih on $(\star)$, we get: $\gamma'(t) \in [\![\tau]\!]_\epsilon$.
By $\Gamma' \subseteq \Gamma$ and $\tau \subseteq \tau'$, we get: $\gamma(t) \in [\![\tau']\!]_\epsilon$, this case is proved.

$\square$

Theorem 0.5 (Preservation). *If* $\cdot \vdash_n t : \tau \wedge t \Downarrow F$ *then* $\vdash_n F : \tau$

Proof. By induction on typing derivation of $\cdot \vdash_n t : \tau$.

Case

$$\frac{}{\cdot \vdash_n c : b}\ \textsc{const}$$

t is c, From E-CONST, we know F is c, It is proved.

For the cases ABS,QUERY,ILAM,VAR,PR, the proof are similar as the one for CONST because t in these cases are values.

Case

$$\frac{\cdot \vdash_{n_1} t_1 : query\ \ (\star)}{\cdot \vdash_{n_1+1} M(t_1) : b}\ \mathrm{MT}$$

t is $M(t_1)$, from the rule E-MECH, we get:

$$\frac{t_1 \Downarrow^m F\ \ (\diamond)}{M(t_1) \Downarrow^m M(F)}\ \textsc{E-mech}$$

By ih on $(\star)$ and $(\diamond)$, we get : $\cdot \vdash_{n_1} F : query$.
Using the rule MT, we conclude $\cdot \vdash_{n_1+1} M(F) : b$. This case is proved.

Case

$$\frac{\cdot \vdash_{n_1} t_1 : \tau_1\ \ (\star) \qquad \cdot \vdash_{n_2} t_2 : \tau_2\ \ (\diamond)}{\cdot \vdash_{\max(n_1,n_2)} (t_1, t_2) : \tau_1 \times \tau_2}\ \textsc{pair}$$

t is (t_1, t_2), from the evaluation rule E-PAIR, we know:

$$\frac{t_1 \Downarrow^{m_1} F_1\ \ (\star\star) \qquad t_2 \Downarrow^{m_2} F_2\ \ (\diamond\diamond)}{(t_1, t_2) \Downarrow^{m_1+m_2} (F_1, F_2)}\ \textsc{E-pair}$$

By ih on $(\star)$ and $(\star\star)$, we get: $\cdot \vdash_{n_1} F_1 : \tau_1$.
By ih on $(\diamond)$ and $(\diamond\diamond)$, we get: $\cdot \vdash_{n_2} F_2 : \tau_2$.
This case is proved by using the rule PAIR.

Case

$$\frac{\cdot \vdash_{n_1} t_1 : \tau_1 \multimap \tau_2\ \ (\star) \qquad \cdot \vdash_{n_2} t_2 : \tau_1\ \ (\diamond)}{\cdot \vdash_{\max(n_1,n_2)} t_1\ t_2 : \tau_2}\ \textsc{app}$$

t is $t_1\ t_2$, from the evaluation rule E-APP, we know:

$$\frac{t_1 \Downarrow^{m_1} \lambda x.t'\ \ (\star\star) \qquad t_2 \Downarrow^{m_2} F\ \ (\diamond\diamond) \qquad t'[F/x] \Downarrow^{m_3} F'\ \ (\heartsuit)}{t_1\ t_2 \Downarrow^{m_1+m_2+m_3} F'}\ \textsc{E-app}$$

By ih on $(\star)$ and $(\star\star)$, we get: $\cdot \vdash_{n_1} \lambda x.t' : \tau_1 \multimap \tau_2\ \ (\spadesuit)$.
By inversion on $(\spadesuit)$, we get: $x : \tau_1 \vdash_{n_1} t' : \tau_2\ (\spadesuit\spadesuit)$.
By ih on $(\diamond)$ and $(\diamond\diamond)$, we get: $\cdot \vdash_{n_2} F : \tau_1\ \ (\clubsuit)$.
From Theorem Substitution with $(\spadesuit\spadesuit)$ and $(\clubsuit)$, we get: $\cdot \vdash_{\max(n_1,n_2)} t'[F/x] : \tau_2\ \ (\heartsuit\heartsuit)$.
By ih on $(\heartsuit)$ and $(\heartsuit\heartsuit)$, we conclude: $\cdot \vdash_{\max(n_1,n_2)} F' : \tau_2$.
It proves this case.

Case

$$\frac{\cdot \vdash_{n_1} t_1 : !_p\tau \ \ (\star) \qquad \cdot, x : [\tau]_p \vdash_{n_2} t_2 : \tau' \ \ (\diamond)}{\cdot \vdash_{\max(n_1,n_2)} \mathsf{let}\ !x = t_1 \ \mathsf{in}\ t_2 : \tau'}\ \text{LET}$$

In this case, t is $\mathsf{let}\ !x = t_1\ \mathsf{in}\ t_2$. From the evaluation rule, we know:

$$\frac{t_1 \Downarrow^{m_1} !t_3 \ \ (\star\star) \qquad t_3 \Downarrow^{m_2} F' \ \ (\diamond\diamond) \qquad t_2[F'/x] \Downarrow^{m_3} F}{\mathsf{let}\ !x = t_1\ \mathsf{in}\ t_2 \Downarrow^{m_1+m_2+m_3} F}\ \text{E-LET-BANG}$$

By ih on $(\star)$ and $(\star\star)$, we get: $\cdot \vdash_{n_1} !t_3 : !_p\tau$ (♠).
By inversion on (♠), we get: $\cdot \vdash_{n_1} t_3 : \tau$ (♠♠).
By ih on (♠♠) and $(\diamond\diamond)$, we get: $\cdot \vdash_{n_2} F' : \tau$ (♣).

Case

$$\frac{\cdot \vdash_{n_1} t' : b \ \ (\star) \qquad \cdot \vdash_{n_2} t_i : b \ \ (\diamond)}{\cdot \vdash_{\max(n_1,n_2)} \mathsf{case}\ t'\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b}\ \text{CASE-CONST}$$

In this case, t is $\mathsf{case}\ t'\ \mathsf{of}\{c_i \Rightarrow t_i\}_{c_i \in b}$.
From the evaluation rule, we get:

$$\frac{t' \Downarrow^{m} F' \ \ (\star\star) \qquad t_i \Downarrow^{m_i} F_i \ \ (\diamond\diamond)}{\mathsf{case}\ t'\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} \Downarrow^{m+m_i} \mathsf{case}\ F'\ \mathsf{of}\ \{c_i \Rightarrow F_i\}_{c_i \in b}}\ \text{E-CASE}$$

By ih on $(\star)$ and $(\star\star)$, we get: $\cdot \vdash_{n_1} F' : b$ (♠).
By ih on $(\diamond)$ and $(\diamond\diamond)$, we get: $\cdot \vdash_{n_2} F_i : b$ (♣).
By the rule CASE-CONST, we conclude :$\cdot \vdash_{\max(n_1,n_2)} \mathsf{case}\ F'\ \mathsf{of}\ \{c_i \Rightarrow F_i\}_{c_i \in b} : b$.
This case is proved.

Case

$$\frac{\cdot \vdash_{n_1} t' : b \ \ (\star) \qquad \cdot \vdash_{n_2} t_i : query \ \ (\diamond)}{\cdot \vdash_{\max(n_1,n_2)} \mathsf{case}\ t'\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : query}\ \text{CASE-QUERY}$$

In this case, t is $\mathsf{case}\ t'\ \mathsf{of}\{c_i \Rightarrow t_i\}_{c_i \in b}$.
From the evaluation rule, we get:

$$\frac{t' \Downarrow^{m} F' \ \ (\star\star) \qquad t_i \Downarrow^{m_i} F_i \ \ (\diamond\diamond)}{\mathsf{case}\ t'\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} \Downarrow^{m+m_i} \mathsf{case}\ F'\ \mathsf{of}\ \{c_i \Rightarrow F_i\}_{c_i \in b}}\ \text{E-CASE}$$

By ih on $(\star)$ and $(\star\star)$, we get: $\cdot \vdash_{n_1} F' : b$ (♠).
By ih on $(\diamond)$ and $(\diamond\diamond)$, we get: $\cdot \vdash_{n_2} F_i : query$ (♣).
By the rule CASE-QUERY, we conclude :$\cdot \vdash_{\max(n_1,n_2)} \mathsf{case}\ F'\ \mathsf{of}\ \{c_i \Rightarrow F_i\}_{c_i \in b} : query$.
This case is proved.

Case

$$\frac{\cdot \vdash_{n_1} t_1 : \tau_1 \times \tau_2 \ \ (\star) \qquad x_1 : \tau_1, x_2 : \tau_2 \vdash_{n_2} t_1' : \tau' \ \ (\diamond)}{\cdot \vdash_{\max(n_1,n_2)} \mathsf{let}\ (x_1, x_2) = t_1\ \mathsf{in}\ t_1' : \tau'}\ \text{LET-P}$$

In this case, t is $\mathsf{let}\ (x_1, x_2) = t_1\ \mathsf{in}\ t_1'$. From the evalutation rule, we get:

$$\frac{t_1 \Downarrow^{m_1} (F_1, F_2) \ \ (\star\star) \qquad t_1'[F_1/x_1][F_2/x_2] \Downarrow^{m_3} F \ \ (\heartsuit)}{\mathsf{let}\ (x_1, x_2) = t_1\ \mathsf{in}\ t_1' \Downarrow^{m_1+m_2+m_3} F}\ \text{E-LET-P}$$

By ih on $(\star)$ and $(\star\star)$, we know: $\cdot \vdash_{n_1} (F_1, F_2) : \tau_1 \times \tau_2$ (♠).
By inversion on (♠), we get : $\cdot \vdash_{n_1'} F_1 : \tau_1$ (♣) and $\cdot \vdash_{n_2'} F_2 : \tau_2$ (♣♣) where $\max(n_1', n_2') = n_1$.
By Theorem Substitution twice with (♣) and (♣♣) on $(\diamond)$, we get $\cdot \vdash_{\max(n_1', \max(n_2', n_2))} t_1'[F_1/x_1][F_2/x_2] : \tau'$ $(\heartsuit\heartsuit)$.

It is proved by ih on (♡♡) and (♡).

Case

$$\frac{\cdot \vdash_{n_1} t_1 : \tau_1 \qquad \cdot \vdash_{n_2} t_2 : \tau_2}{\cdot \vdash_{\max(n_1,n_2)} (t_1, t_2) : \tau_1 \times \tau_2} \text{ PAIR}$$

In this case, t is (t_1, t_2). From the evaluation rule we get :

$$\frac{t_1 \Downarrow^{m_1} F_1 \qquad t_2 \Downarrow^{m_2} F_2}{(t_1, t_2) \Downarrow^{m_1+m_2} (F_1, F_2)} \text{ E-PAIR}$$

By ih, we know : $\cdot \vdash_{n_1} F_1 : \tau_1$ and $\cdot \vdash_{n_2} F_2 : \tau_2$.
This case is proved by using the rule PAIR.

Case

$$\frac{\Gamma \vdash_n t : \tau \;\; (\star) \qquad \Gamma' \subseteq \Gamma \qquad \vDash n \leq n' \qquad \tau \subseteq \tau'}{\Gamma' \vdash_{n'} t : \tau'} \text{ SUB}$$

by ih on ($\star$), we get: $\Gamma \vdash_n F : \tau$.
This case is proved by using the rule SUB.

□

Theorem 0.6 (Substitution). *If* $\Gamma \vdash_{n_1} t_1 : \tau_1$ *and* $\Delta, x : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2$ *then* $\max(\Gamma, \Delta) \vdash_{\max(p+n_1,n_2)} t_2[t_1/x] : \tau_2$

Proof. The theorem is proved by induction on the typing derivation of the second premise $\Gamma, x : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2$.

Case

$$\frac{}{\Gamma, x : [\tau_1]_p \vdash_0 x : \tau_1}\ \text{VAR}$$

where $t_2 = x$ and $\tau_2 = \tau_1$.
Assume we know: $\Gamma \vdash_{n_1} t_1 : \tau_1 \quad (\star)$
TS: $\Gamma \vdash_{\max(p+n_1,0)} t_2[t_1/x] : \tau_1$
STS: $\Gamma \vdash_{\max(p+n_1,0)} x[t_1/x] : \tau_1 \Rightarrow \Gamma \vdash_{\max(p+n_1,0)} t_1 : \tau_1$
It is proved by using Lemma **??** on $(\star)$ because $n_1 \leq \max(p + n_1, 0)$.

Case

$$\frac{}{\Gamma, y : [\tau_2]_{p'}, x : [\tau_1]_p \vdash_0 y : \tau_2}\ \text{VAR}$$

Assume we know: $\Gamma, y : [\tau_2]_{p'} \vdash_{n_1} t_1 : \tau_1 \quad (\star)$
TS: $\Gamma, y : [\tau_2]_{p'} \vdash_{\max(p+n_1,0)} y[t_1/x] : \tau_2$
STS: $\Gamma, y : [\tau_2]_{p'} \vdash_{\max(p+n_1,0)} y : \tau_2$
From VAR, we get : $\Gamma, y : [\tau_2]_{p'} \vdash_0 y : \tau_2 \ (\diamond)$. It is proved by using Lemma **??** on $(\diamond)$.

Case

$$\frac{\Gamma, x : [\tau_1]_p \vdash_n t : query \quad (\star\star)}{1 + \Gamma, x : [\tau_1]_{p+1} \vdash_{n+1} M(t) : b}\ \text{MT}$$

Assume we know: $1 + \Gamma \vdash_{n_1} t_1 : \tau_1 \quad (\star)$
We know that $\Gamma \vdash_{n_1} t_1 : \tau_1$ by Lemma **??** on the assumption $(\star)$.
We also know that $n + 1 = n_2$.
TS: $1 + \Gamma \vdash_{\max(p+n_1+1,n_2)} (M(t))[t_1/x] : b$
By IH on $(\star\star)$, we know: $\Gamma \vdash_{\max(n_1+p,n)} t[t_1/x] : query \quad (\diamond)$
By rule MT, we know that :

$$\frac{\Gamma \vdash_{\max(n_1+p,n)} t[t_1/x] : query}{1 + \Gamma \vdash_{1+\max(n_1+p,n)} M(t[t_1/x]) : b}\ \text{MT}$$

We obtain: $1 + \Gamma \vdash_{\max(n_1+p+1,n+1)} M(t[t_1/x]) : b \Rightarrow 1 + \Gamma \vdash_{\max(n_1+p+1,n_2)} \big(M(t)\big)[t_1/x] : b$
This case is proved.

Case

$$\frac{\Gamma, x : [\tau_1]_p \vdash_{n_2} t : \tau \quad (\star\star)}{p' + \Gamma, x : [\tau_1]_{p+p'} \vdash_{n_2} !t : !_{p'}\tau}\ \text{PR}$$

We assume $p' + \Gamma \vdash_{n_1} t_1 : \tau_1$. by Lemma **??**, we know $\Gamma \vdash_{n_1} t_1 : \tau_1$ $(\star)$.
TS: $p' + \Gamma \vdash_{\max(n_1+p+p',n_2)} !t[t_1/x] : !_{p'}\tau$

By IH on $(\star\star)$ along with $(\star)$, we know: $\Gamma \vdash_{\max(n_1+p,n_2)} t[t_1/x] : \tau \quad (\diamond)$
By rule PR, we know:

$$\frac{\Gamma \vdash_{\max(n_1+p,n_2)} t[t_1/x] : \tau}{p' + \Gamma \vdash_{\max(n_1+p,n_2)} !t[t_1/x] : !_{p'}\tau \quad (\diamond\diamond)}\ \text{PR}$$

This case is proved by Lemma **??** on $(\diamond\diamond)$.

Case

$$\frac{}{\Gamma, x : [\tau_1]_p \vdash_0 q : query}\ \text{QUERY}$$

TS: $\Gamma \vdash_{\max(p+n_1,n_2)} q[t_1/x] : query \Rightarrow \Gamma \vdash_{\max(p+n_1,n_2)} q : query$.
Using rule QUERY, we get $\Gamma \vdash_0 q : QUERY$, This case is proved by Lemma **??** on it.

Case

$$\frac{\Gamma_1, y : [\tau_1]_p \vdash_n t : !_{p_1}\tau \quad (\diamond) \qquad \Gamma_2, y : [\tau_1]_p, x : [\tau]_{p_1} \vdash_{n'} t' : \tau' \quad (\diamond\diamond)}{\max(\Gamma_1, \Gamma_2), y : [\tau_1]_p \vdash_{\max(n,n')} \mathsf{let}\ !x = t \ \mathsf{in}\ t' : \tau'}\ \text{LET}$$

We assume $\max(\Gamma_1, \Gamma_2) \vdash_{n_1} t_1 : \tau$ $(\star)$.

TS: $\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1+p, \max(n,n'))} (\mathsf{let}\ !x = t \ \mathsf{in}\ t')[t_1/y] : \tau'$
By Lemma **??**, Lemma **??**, we can extend the context of $(\diamond)$, to $\max(\Gamma_1, \Gamma_2)$:
$\max(\Gamma_1, \Gamma_2), y : [\tau_1]_p \vdash_n t : !_{p_1}\tau (\diamond')$.
Similarly, from $(\diamond\diamond)$we get: $\max(\Gamma_1, \Gamma_2), y : [\tau_1]_p, x : [\tau]_{p_1} \vdash_{n'} t' : \tau'$ $(\diamond\diamond')$.
By IH on $(\diamond')$, we get: $\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1+p,n)} t[t_1/y] : !_{p_1}\tau$ $(\spadesuit)$
By IH on $(\diamond\diamond')$, we get: $\max(\Gamma_1, \Gamma_2), x : [\tau]_{p_1} \vdash_{\max(n_1+p,n')} t'[t_1/y] : \tau'$ $(\clubsuit)$
By rule let, we get:

$$\frac{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1+p,n)} t[t_1/y] : !_{p_1}\tau \quad (\spadesuit) \qquad \max(\Gamma_1, \Gamma_2), x : [\tau]_{p_1} \vdash_{\max(n_1+p,n')} t'[t_1/y] : \tau' \quad (\clubsuit)}{\Gamma \vdash_{\max(\max(n_1+p,n), \max(n_1+p,n'))} \mathsf{let}\ !x = t[t_1/y] \ \mathsf{in}\ t'[t_1/y] : \tau'}\ \text{LET}$$

Because $\max(\max(n_1+p, n), \max(n_1+p, n')) = \max(n_1+p, \max(n, n'))$, this case is proved.

Case

$$\frac{\Gamma, y : [\tau]_p, x : [\tau_1]_0 \vdash_n t : \tau_2 \ (\diamond)}{\Gamma, y : [\tau]_p \vdash_n \lambda x.t : \tau_1 \multimap \tau_2}\ \text{ABS}$$

We assume $\Gamma \vdash_{n_1} t_1 : \tau$ $(\star)$.
By Lemma **??**, we get: $\Gamma, x : [\tau_1]_0 \vdash_{n_1} t_1 : \tau_1$.
TS: $\Gamma \vdash_{\max(n_1+p,n)} \lambda x.t[t_1/y] : \tau_1 \multimap \tau_2$.
By IH on $(\diamond)$, we get: $\Gamma, x : [\tau_1]_0 \vdash_{\max(n_1+p,n)} t[t_1/y] : \tau_2$ $(\spadesuit)$.
It is proved by the rule ABS and $(\spadesuit)$.
Case

$$\frac{\Gamma_1, x : [\tau]_p \vdash_{n_1} t_1 : \tau_1 \to \tau_2 \ \ (\diamond) \qquad \Gamma_2, x : [\tau]_p \vdash_{n_2} t_2 : \tau_1 \ \ (\diamond\diamond)}{\max(\Gamma_1, \Gamma_2), x : [\tau]_p \vdash_{\max(n_1,n_2)} t_1\ t_2 : \tau_2}\ \text{APP}$$

We assume $\max(\Gamma_1, \Gamma_2) \vdash_n t : \tau$.
By Lemma **??**, Lemma **??**, we extend the
Case

$$\frac{\Gamma_1 \vdash_{n_1} t : \tau_1 \times \tau_2 \qquad \Gamma_2, x_1 : \tau_1, x_2 : \tau_2 \vdash_{n_2} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ (x_1, x_2) = t \ \mathsf{in}\ t' : \tau'}\ \text{LET-P}$$

□

Theorem 0.7 (Substitution)**.**

1. *If* $\Gamma \vdash_{n_1} t_1 : \tau_1$ *and* $\Delta, x : \tau_1 \vdash_{n_2} t_2 : \tau_2$ *then* $max(\Gamma, \Delta) \vdash_{\max(n_1, n_2)} t_2[t_1/x] : \tau_2$
2. *If* $\Gamma \vdash_{n_1} t_1 : !_p \tau_1$ *and* $\Delta, x : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2$ *then* $max(\Gamma, \Delta) \vdash_{\max(n_1+p, n_2)} t_2[t_1/x] : \tau_2$

$$\begin{array}{ccc}
c[t_1/x] & ::= & c \\
M(t)[t_1/x] & & M(t[t_1/x]) \\
x[t_1/x] & & t_1 \\
y[t_1/x] & & y \\
(t\,t')[t_1/x] & & t[t_1/x]\,t'[t_1/x] \\
(\lambda y.t)[t_1/x] & & \lambda y.t[t_1/y] \\
(\mathsf{let}\ y = t\ \mathsf{in}\ t')[t_1/x] & & \mathsf{let}\ y = t[t_1/x]\ \mathsf{in}\ t'[t_1/x]
\end{array}$$

Proof. of 0.7.1
The theorem is proved by induction on the typing derivation of the second premise $\Delta, x : \tau_1 \vdash_{n_2} t_2 : \tau_2$.
Assume we know $\Gamma \vdash_{n_1} t_1 : \tau_1$ $(\diamond)$.
Case

$$\frac{}{\Delta, x : \tau_1 \vdash_n c : b \ (\star)}\ \text{CONST}$$

where $t_2 = c$, $\tau_2 = b$ and $n_2 = n$.
To show: $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} c[t_1/x] : b$.
Because $c[t_1/x] = c$, it suffices to show $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} c : b$.
This case is proved by applying the CONST rule.
Case

$$\frac{}{\Delta, x : \tau_1 \vdash_n q : query \ (\star)}\ \text{QUERY}$$

where $t_2 = q$, $\tau_2 = query$ and $n_2 = n$.
To show: $\max \Gamma, \Delta \vdash_{\max(n_1, n)} q[t_1/x] : query$.
Because $q[t_1/x] = q$, it suffices to show: $max(\Gamma, \Delta) \vdash_{\max(n_1, n)} q : query$.
This case is proved by applying the QUERY rule.
Case
SubCase 1

$$\frac{}{\Delta, x : \tau_1 \vdash_n x : \tau_1}\ \text{VAR}$$

where $t_2 = x$, $\tau_2 = \tau_1$ and $n_2 = n$.
To show $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} x[t_1/x] : \tau_1$.
it suffices to show $\max(\Gamma, \Delta) \vdash_{\max(n_1, n_2)} t_1 : \tau_1$.
This case is proved by applying Lemma 5 and 6 on $(\diamond)$.
SubCase 2

$$\frac{}{\Delta, x : \tau_1, y : \tau_2 \vdash_n y : \tau_2}\ \text{VAR}$$

where $t_2 = y$ and $n_2 = n$.
To show $\max(\Gamma, \Delta, y : \tau_2) \vdash_{\max(n_1, n)} y[t_1/x] : \tau_2$.
it suffices to show $\max(\Gamma, \Delta, y : \tau_2) \vdash_n y : \tau_2$.
This case is proved by applying the rule VAR.
Case

$$\frac{\Delta, x : \tau_1, y : \tau_1' \vdash_n t : \tau_2' \ (\star)}{\Delta, x : \tau_1 \vdash_n \lambda y.t : \tau_1' \multimap \tau_2'}\ \text{ABS}$$

where $t_2 = \lambda y.t$, $\tau_2 = \tau_1' \multimap \tau_2'$ and $n_2 = n$.
To show $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} \lambda y.t\, [t_1/x] : \tau_1' \multimap \tau_2'$.
By applying Lemma 7 and induction hypothesis on ($\star$), we get:
$\max(\Gamma, \Delta, y : \tau_1') \vdash_{\max(n_1, n_2)} t[t_1/x] : \tau_2'$ ($\star\star$).
By applying the ABS rule on ($\star\star$), we get: $\max(\Gamma, \Delta) \vdash_{\max(n_1, n_2)} \lambda y.(t[t_1/x]) : \tau_1' \multimap \tau_2'$, this case is proved.

Case

$$\frac{[\Delta_2] \vdash_n t : query \ \ (\star)}{\Delta_1, 1 + [\Delta_2], x : \tau_1 \vdash_{n+1} M(t) : b} \text{MT}$$

Where $\Delta = \Delta_1, 1 + [\Delta_2]$, $t_2 = M(t)$, $\tau_2 = b$ and $n_2 = n + 1$.
To show $\max(\Gamma, \Delta_1, 1 + [\Delta_2]) \vdash_{\max(n_1, n+1)} M(t)[t_1/x] : b$.
We know $x \notin \text{dom}([\Delta_2]) \wedge [\Delta_2] \vdash_n t : query \implies x \notin \text{FV}(t)$.
So we know $t = t[t_1/x] \implies [\Delta_2] \vdash_n t[t_1/x] : query$ ($\star\star$)
By applying the MT rule on ($\star\star$) we get:
$\max(\Delta_1, 1 + [\Delta_2]) \vdash_{1+n} M(t)[t_1/x] : b$ ($\clubsuit$)
This case is proved using Lemma 7, Lemma 3 on ($\clubsuit$).

Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau_1 \to \tau_2 \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau_1}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1', n_2')} t_1'\, t_2' : \tau_2} \text{APP}$$

There are three sub cases depending on whether $x \in \text{dom}(\Gamma_1)$ and $\text{dom}(\Gamma_2)$.

SubCase 1

$$\frac{\Delta_1, x : \tau_1 \vdash_{n_1'} t_1' : \tau_1 \to \tau_2 \ \ (\star) \qquad \Delta_2 \vdash_{n_2'} t_2' : \tau_1 \ \ (\clubsuit)}{\max(\Delta_1, x : \tau_1, \Delta_2) \vdash_{\max(n_1', n_2')} t_1'\, t_2' : \tau_2} \text{APP}$$

where $x \notin \text{dom}(\Delta_2)$.
$\Delta = \max(\Delta_1, \Delta_2)$, $t_2 = t_1'\, t_2'$ and $n_2 = \max(n_1', n_2')$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n_1', n_2'))} t_1'\, t_2'\, [t_1/x] : \tau_2$.
it suffices to show $\max(\Gamma, \Delta_1, \Delta_2) \vdash_{\max(n_1, n_1', n_2')} t_1'\, t_2'\, [t_1/x] : \tau_2$.
By induction hypothesis on ($\star$), we get: $\max(\Gamma, \Delta_1) \vdash_{\max(n_1, n_1')} t_1'[t_1/x] : \tau_1 \to \tau_2$ ($\star\star$).
Because $x \notin \text{dom}(\Delta_2) \implies t_2'[t_1/x] = t_2'$.
By applying APP on ($\star\star$) and ($\clubsuit$), we get: $\max(\max(\Gamma, \Delta_1), \Delta_2) \vdash_{\max(\max(n_1, n_1'), n_2')} (t_1'\, t_2')[t_1/x] : \tau_2$.
This case is proved.

SubCase 2

$$\frac{\Delta_1 \vdash_{n_1'} t_1' : \tau_1 \to \tau_2 \ \ (\star) \qquad \Delta_2, x : \tau_1 \vdash_{n_2'} t_2' : \tau_1 \ \ (\clubsuit)}{\max(\Delta_1 \Delta_2, x : \tau_1) \vdash_{\max(n_1', n_2')} t_1'\, t_2' : \tau_2} \text{APP}$$

where $x \notin \text{dom}(\Delta_1)$.
This case is proved by induction hypothesis on ($\clubsuit$) and applying APP rule.

SubCase 3

$$\frac{\Delta_1, x : \tau_1 \vdash_{n_1'} t_1' : \tau_1 \to \tau_2 \ \ (\star) \qquad \Delta_2, x : \tau_1 \vdash_{n_2'} t_2' : \tau_1 \ \ (\clubsuit)}{\max(\Delta_1, \Delta_2, x : \tau_1) \vdash_{\max(n_1', n_2')} t_1'\, t_2' : \tau_2} \text{APP}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n_1', n_2'))} t_1'\, t_2' : \tau_2$.
This case is proved by induction hypothesis on both ($\clubsuit$) and ($\star$) and then applying APP rule.

Case

$$\frac{\Delta', x:\tau_1, y:\tau \vdash_n t_2 : \tau_2 \ \ (\star)}{\Delta', x:\tau_1, y:[\tau]_p \vdash_n t_2 : \tau_2}\ \text{DER}$$

where $\Delta = \Delta', y:[\tau]_p$, $n_2 = n$.
To show $\max(\Gamma, \Delta', y:[\tau]_p) \vdash_{\max(n_1,n)} t_2[t_1/x] : \tau_2$.
By induction hypothesis on $(\star)$, we get: $\max(\Gamma, \Delta', y:\tau) \vdash_{\max(n_1,n)} t_2[t_1/x] : \tau_2$ $(\star\star)$.
By applying the DER rule on $(\star\star)$, we get: $\max(\Gamma, \Delta', y:[\tau]_p) \vdash_{\max(n_1,n)} t_2[t_1/x] : \tau_2$.
This case is proved.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : !_p\tau \qquad \Gamma_2, y:[\tau]_p \vdash_{n_2} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} \mathsf{let}\ !y = t\ \mathsf{in}\ t' : \tau'}\ \text{LET}$$

There are three sub cases.

Subcase 1: x $\notin$ **dom**(Δ_2)

$$\frac{\Delta_1, x:\tau_1 \vdash_{n_1'} t : !_p\tau\ \ (\star) \qquad \Delta_2, y:[\tau]_p \vdash_{n_2'} t' : \tau'\ \ (\clubsuit)}{\max(\Delta_1, x:\tau_1, \Delta_2) \vdash_{\max(n_1',n_2')} \mathsf{let}\ !y = t\ \mathsf{in}\ t' : \tau'}\ \text{LET}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1,\max(n_1',n_2'))} (\mathsf{let}\ !y = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By IH on $(\star)$, we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1,n_1')} t[t_1/x] : !_p\tau$ $(\star\star)$.
$x \notin \mathrm{dom}(\Delta_2) \implies t'[t_1/x] = t'$.
It is proved by the rule LET using $(\star\star)$ and $(\clubsuit)$.

Subcase 2: x $\notin$ **dom**(Δ_1)

$$\frac{\Delta_1 \vdash_{n_1'} t : !_p\tau\ \ (\star) \qquad \Delta_2, x:\tau_1, y:[\tau]_p \vdash_{n_2'} t' : \tau'\ \ (\clubsuit)}{\max(\Delta_1, x:\tau_1, \Delta_2) \vdash_{\max(n_1',n_2')} \mathsf{let}\ !y = t\ \mathsf{in}\ t' : \tau'}\ \text{LET}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1,\max(n_1',n_2'))} (\mathsf{let}\ !y = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By IH on $(\clubsuit)$, we get $\max(\Gamma, \Delta_2), y:[\tau]_p \vdash_{\max(n_1,n_2')} t'[t_1/x] : \tau'$ $(\clubsuit\clubsuit)$.
$x \notin \mathrm{dom}(\Delta_1) \implies t[t_1/x] = t$.
It is proved by the rule LET using $(\star)$ and $(\clubsuit\clubsuit)$.

Subcase 3

$$\frac{\Delta_1, x:\tau_1 \vdash_{n_1'} t : !_p\tau\ \ (\star) \qquad \Delta_2, x:\tau_1, y:[\tau]_p \vdash_{n_2'} t' : \tau'\ \ (\clubsuit)}{\max(\Delta_1, x:\tau_1, \Delta_2) \vdash_{\max(n_1',n_2')} \mathsf{let}\ !y = t\ \mathsf{in}\ t' : \tau'}\ \text{LET}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1,\max(n_1',n_2'))} (\mathsf{let}\ !y = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By IH on $(\star)$, we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1,n_1')} t[t_1/x] : !_p\tau$ $(\star\star)$.
By IH on $(\clubsuit)$, we get $\max(\Gamma, \Delta_2), y:[\tau]_p \vdash_{\max(n_1,n_2')} t'[t_1/x] : \tau'$ $(\clubsuit\clubsuit)$.
It is proved by the rule LET using $(\star\star)$ and $(\clubsuit\clubsuit)$.

Case

$$\frac{\Gamma_1 \vdash_{n_1'} t : \tau_1' \times \tau_2' \qquad \Gamma_2, x_1:\tau_1', x_2:\tau_2' \vdash_{n_2'} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1',n_2')} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'}\ \text{LET-P}$$

There are three sub cases.
Subcase 1: x $\notin$ **dom**(Δ_2)

$$\frac{\Delta_1, x:\tau_1 \vdash_{n_1'} t : \tau_1' \times \tau_2'\ \ (\star) \qquad \Delta_2, x_1:\tau_1', x_2:\tau_2' \vdash_{n_2'} t' : \tau'\ \ (\clubsuit)}{\max(\Delta_1, x:\tau_1, \Delta_2) \vdash_{\max(n_1',n_2')} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'}\ \text{LET-P}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n_1', n_2'))} (\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By IH on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1, n_1')} t[t_1/x] : \tau_1' \times \tau_2'$ ($\star\star$).
$x \notin \mathrm{dom}(\Delta_2) \implies t'[t_1/x] = t'$.
It is proved by the rule LET-P using ($\star\star$) and ($\clubsuit$).

Subcase 2: x $\notin$ dom(Δ_1)

$$\frac{\Delta_1 \vdash_{n_1'} t : \tau_1' \times \tau_2' \ (\star) \qquad \Delta_2, x_1 : \tau_1', x_2 : \tau_2', x : \tau_1 \vdash_{n_2'} t' : \tau' \ (\clubsuit)}{\max(\Delta_1, x : \tau_1, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'} \text{ LET-P}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n_1', n_2'))} (\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By IH on ($\clubsuit$), we get $\max(\Gamma, \Delta_2), x_1 : \tau_1', x_2 : \tau_2' \vdash_{\max(n_1, n_2')} t'[t_1/x] : \tau'$ ($\clubsuit\clubsuit$).
$x \notin \mathrm{dom}(\Delta_1) \implies t[t_1/x] = t$.
It is proved by the rule LET-P using ($\star$) and ($\clubsuit\clubsuit$).

Subcase 3

$$\frac{\Delta_1, x : \tau_1 \vdash_{n_1'} t : \tau_1' \times \tau_2' \ (\star) \qquad \Delta_2, x_1 : \tau_1', x_2 : \tau_2', x : \tau_1 \vdash_{n_2'} t' : \tau' \ (\clubsuit)}{\max(\Delta_1, x : \tau_1, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'} \text{ LET-P}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n_1', n_2'))} (\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By IH on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1, n_1')} t[t_1/x] : \tau_1' \times \tau_2'$ ($\star\star$).
By IH on ($\clubsuit$), we get $\max(\Gamma, \Delta_2), x_1 : \tau_1', x_2 : \tau_2' \vdash_{\max(n_1, n_2')} t'[t_1/x] : \tau'$ ($\clubsuit\clubsuit$).
It is proved by the rule LET-P using ($\star\star$) and ($\clubsuit\clubsuit$).

Case

$$\frac{\Gamma_1 \vdash_{n_1'} t_1 : \tau_1 \qquad \Gamma_2 \vdash_{n_2'} t_2 : \tau_2}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1', n_2')} (t_1, t_2) : \tau_1 \times \tau_2} \text{ PAIR}$$

There are three sub cases,x only in Γ_1, x only in Γ_2 and x appears in both Γ_1 and Γ_2. When x only in Γ_1, it is proved by ih on the first premise and then using the rule PAIR. When x only in Γ_2, it is proved by ih on the second premise and then using the rule PAIR. When x appears in both Γ_1 and Γ_2, it is proved by ih on both premises and then using rule PAIR.

Case

$$\frac{\Gamma_1 \vdash_{n_1'} t : b \qquad \Gamma_2 \vdash_{n_2'} t_i : b}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1', n_2')} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \text{ CASE-CONST}$$

There are three sub cases.

Subcase 1: x $\notin$ dom(Δ_2)

$$\frac{\Delta_1, x : \tau_1 \vdash_{n_1'} t : b \ (\star) \qquad \Delta_2 \vdash_{n_2'} t_i : b \ (\clubsuit)}{\max(\Delta_1, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \text{ CASE-CONST}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n_1', n_2'))} (\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b})[t_1/x] : b$.
By IH on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1, n_1')} t[t_1/x] : b$ ($\star\star$).
$x \notin \mathrm{dom}(\Delta_2) \implies t_i[t_1/x] = t_i$.
It is proved by the rule CASE-CONST using ($\star\star$) and ($\clubsuit$).

Subcase 2: $\mathbf{x} \not\in \mathbf{dom}(\Delta_1)$

$$\frac{\Delta_1 \vdash_{n'_1} t : b \ \ (\star) \qquad \Delta_2, x : \tau_1 \vdash_{n'_2} t_i : b \ \ (\clubsuit)}{\max(\Delta_1, \Delta_2) \vdash_{\max(n'_1, n'_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \ \textsc{case-const}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n'_1, n'_2))} (\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b})[t_1/x] : b$.
By IH on ($\clubsuit$), we get $\max(\Gamma, \Delta_2) \vdash_{\max(n_1, n'_2)} t_i[t_1/x] : b \ \ (\clubsuit\clubsuit)$.
$x \not\in \mathrm{dom}(\Delta_1) \implies t[t_1/x] = t$.
It is proved by the rule CASE-CONST using ($\star$) and ($\clubsuit\clubsuit$).

Subcase 3

$$\frac{\Delta_1, x : \tau_1 \vdash_{n'_1} t : b \ \ (\star) \qquad \Delta_2, x : \tau_1 \vdash_{n'_2} t_i : b \ \ (\clubsuit)}{\max(\Delta_1, \Delta_2) \vdash_{\max(n'_1, n'_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \ \textsc{case-const}$$

TS: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1, \max(n'_1, n'_2))} (\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b})[t_1/x] : b$.
By IH on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1, n'_1)} t[t_1/x] : b \ \ (\star\star)$.
By IH on ($\clubsuit$), we get $\max(\Gamma, \Delta_2) \vdash_{\max(n_1, n'_2)} t_i[t_1/x] : b \ \ (\clubsuit\clubsuit)$..
It is proved by the rule CASE-CONST using ($\star\star$) and ($\clubsuit\clubsuit$).

Case

$$\frac{\Gamma_1 \vdash_{n'_1} t : b \qquad \Gamma_2 \vdash_{n'_2} t_i : query}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n'_1, n'_2)} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : query} \ \textsc{case-query}$$

There are three sub cases. Thre proof are quite similar as the one of *CASE-CONST*.

Case

$$\frac{i :: \mathbb{N}; \Delta, x : \tau_1 \vdash_n t : \tau \ \ (\star) \qquad i \notin \mathrm{FIV}(\Delta, x : \tau_1)}{\Delta, x : \tau_1 \vdash_n \Lambda.t : \forall i{::}\mathbb{N}.\tau} \ \textsc{iabs}$$

TS: $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} \Lambda.t[t_1/x] : \forall i{::}\mathbb{N}.\tau$.
By ih on ($\star$), we get : $i :: \mathbb{N}; \max(\Gamma, \Delta) \vdash_{\max(n_1, n)} t[t_1/x] : \tau$ ($\star\star$).
There are two cases.
Sub case 1: $i \not\in \mathbf{FIV}(\Gamma)$ It is proved by using rule IABS with ($\star\star$).
Sub case 2: $i \in \mathbf{FIV}(\Gamma)$ We choose $j \not\in \mathrm{FIV}(\max(\Gamma, \Delta))$. We rename all the i to j in ($\star\star$) and use rule IABS to get: $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} \Lambda.t[t_1/x] : \forall j{::}\mathbb{N}.\tau$. It is just a renaming version of the goal.
Case

$$\frac{\Delta, x : \tau_1 \vdash_n t : \forall i{::}\mathbb{N}.\tau \ (\star) \qquad \vdash I :: \mathbb{N}}{\Delta, x : \tau_1 \vdash_n t\,[] : \tau\{I/i\}} \ \textsc{iapp}$$

TS: $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} t[t_1/x]\,[] : \tau\{I/i\}$.
By ih on ($\star$), we get: $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} t : \forall i{::}\mathbb{N}.\tau \ \ (\star\star)$.
It is proved by the rule IAPP with ($\star\star$).

Case

$$\frac{\Delta, x : \tau_1 \vdash_n t : \tau \ \ (\star) \qquad \Delta', x : \tau_1 \subseteq \Delta, x : \tau_1 \ (\clubsuit) \qquad \vDash n \leq n' \ (\spadesuit) \qquad \tau \subseteq \tau'}{\Delta', x : \tau_1 \vdash_{n'} t : \tau'} \ \textsc{sub}$$

TS: $\max(\Gamma, \Delta') \vdash_{\max(n', n_1)} t : \tau'$.
By ih on ($\star$), we get : $\max(\Gamma, \Delta) \vdash_{\max(n, n_1)} t : \tau$.
($\clubsuit$) $\implies \max(\Gamma, \Delta') \subseteq \max(\Gamma, \Delta)$ ($\clubsuit\clubsuit$).
($\spadesuit$) $\implies \vDash \max(n_1, n) \leq \max(n_1, n')$.
It is proved by the rule SUB.

Case

$$\frac{[\Delta_2] \vdash_n t : \tau}{\Delta_1, p + [\Delta_2], x : \tau_1 \vdash_{n+p} !t : !_p \tau} \text{ PR}$$

TS: $\max(\Gamma, (\Delta_1, p + [\Delta_2])) \vdash_{\max(n_1, n+p)} !t[t_1/x] : !_p \tau$.
$x \notin [\Delta_2] \implies [\Delta_2] \vdash_n t[t_1/x] : \tau$ $(\star)$.
By rule PR with $(\star)$, we get: $\Delta_1, p + [\Delta_2] \vdash_{n+p} !t[t_1/x] : !_p \tau$. It is proved by the Lemma 7 and Lemma 3 from the conclusion.

□

Proof. of 0.7.2
The theorem is proved by induction on the typing derivation of the second premise $\Delta, x : [\tau_1]_p \vdash_{n_2} t_2 : \tau_2$.
Assume we know $\Gamma \vdash_{n_1} t_1 : !_p \tau_1 \ (\diamond)$.

$$\frac{}{\Delta, x : [\tau_1]_p \vdash_n c : b \ (\star)} \text{CONST}$$

where $t_2 = c$, $\tau_2 = b$ and $n_2 = n$.
To show: $\max(\Gamma, \Delta) \vdash_{\max(n_1+p,n)} c[t_1/x] : b$.
Because $c[t_1/x] = c$, it suffices to show $\max(\Gamma, \Delta) \vdash_{\max(n_1+p,n)} c : b$.
This case is proved by applying the CONST rule.
Case

$$\frac{}{\Delta, x : [\tau_1]_p \vdash_n q : query \ (\star)} \text{QUERY}$$

where $t_2 = q$, $\tau_2 = query$ and $n_2 = n$.
To show: $\max \Gamma, \Delta \vdash_{\max(n_1+p,n)} q[t_1/x] : query$.
Because $q[t_1/x] = q$, it suffices to show: $max(\Gamma, \Delta) \vdash_{\max(n_1,n)} q : query$.
This case is proved by applying the QUERY rule.
Case

$$\frac{}{\Delta, x : [\tau_1]_p, y : \tau_2 \vdash_n y : \tau_2} \text{VAR}$$

where $t_2 = y$ and $n_2 = n$.
To show $\max(\Gamma, \Delta, y : \tau_2) \vdash_{\max(n_1+p,n)} y[t_1/x] : \tau_2$.
it suffices to show $\max(\Gamma, \Delta, y : \tau_2) \vdash_n y : \tau_2$.
This case is proved by applying the rule VAR.
Case

$$\frac{\Delta, x : [\tau_1]_p, y : \tau'_1 \vdash_n t : \tau'_2 \ (\star)}{\Delta, x : [\tau_1]_p \vdash_n \lambda y.t : \tau'_1 \multimap \tau'_2} \text{ABS}$$

where $t_2 = \lambda y.t$, $\tau_2 = \tau'_1 \multimap \tau'_2$ and $n_2 = n$.
To show $\max(\Gamma, \Delta) \vdash_{\max(n_1+p,n)} \lambda y.t\,[t_1/x] : \tau'_1 \multimap \tau'_2$.
By applying Lemma 7 and induction hypothesis on $(\star)$, we get:
$\max(\Gamma, \Delta, y : \tau'_1) \vdash_{\max(n_1+p,n_2)} t[t_1/x] : \tau'_2 \ (\star\star)$.
By applying the ABS rule on $(\star\star)$, we get: $\max(\Gamma, \Delta) \vdash_{\max(n_1+p,n_2)} \lambda y.(t[t_1/x]) : \tau'_1 \multimap \tau'_2$, this case is proved.

Case

$$\frac{[\Delta_2], x : [\tau_1]_{p-1} \vdash_n t : query \ (\star)}{\Delta_1, 1 + [\Delta_2], x : [\tau_1]_p \vdash_{n+1} M(t) : b} \text{MT}$$

Where $\Delta = \Delta_1, 1 + [\Delta_2]$, $t_2 = M(t)$, $\tau_2 = b$ and $n_2 = n + 1$.
To show $max(\Gamma, \Delta_1, 1 + [\Delta_2]) \vdash_{\max(n_1+p,n+1)} M(t)[t_1/x] : b$.
By induction hypothesis on $(\star)$, we get: $max(\Gamma, [\Delta_2]) \vdash_{\max(n_1+p-1,n)} t[t_1/x] : query \ (\star\star)$.
By applying the MT rule on $(\star\star)$, we get: $\Delta_1, max(\Gamma, 1 + [\Delta_2]) \vdash_{1+\max(n_1+p-1,n)} M(t[t_1/x]) : b$.
Because Δ_1 and Δ_2 are disjoint and $1 + \max(n_1 + p - 1, n) = \max(n_1 + p, n + 1)$, we get:
$\max(\Gamma, (\Delta_1, 1 + [\Delta_2])) \vdash_{\max(n_1+p,n+1)} M(t)[t_1/x] : b$.
This case is proved.
Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau_1 \to \tau_2 \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau_1}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1,n_2)} t_1\, t_2 : \tau_2} \text{APP}$$

There are three sub cases depending on whether $x \in \mathrm{dom}(\Gamma_1)$ and $\mathrm{dom}(\Gamma_2)$.

Subcase 1 x $\notin$ dom(Δ_2)

$$\frac{\Delta_1, x:[\tau_1]_p \vdash_{n_1'} t_1' : \tau_1 \to \tau_2 \ (\star) \qquad \Delta_2 \vdash_{n_2'} t_2' : \tau_1 \ (\clubsuit)}{\max(\Delta_1, x:[\tau_1]_p, \Delta_2) \vdash_{\max(n_1', n_2')} t_1' \, t_2' : \tau_2} \text{ APP}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 = t_1' \, t_2'$ and $n_2 = \max(n_1', n_2')$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n_1', n_2'))} t_1' \, t_2' \, [t_1/x] : \tau_2$.
it suffices to show $\max(\Gamma, \Delta_1, \Delta_2) \vdash_{\max(n_1+p, n_1', n_2')} t_1' \, t_2' \, [t_1/x] : \tau_2$.
By induction hypothesis on $(\star)$, we get: $\max(\Gamma, \Delta_1) \vdash_{\max(n_1+p, n_1')} t_1'[t_1/x] : \tau_1 \to \tau_2 \ (\star\star)$.
Because $x \notin \mathrm{dom}(\Delta_2) \implies t_2'[t_1/x] = t_2'$.
By applying APP on $(\star\star)$ and $(\clubsuit)$, we get: $\max(\max(\Gamma, \Delta_1), \Delta_2) \vdash_{\max(\max(n_1+p, n_1'), n_2')} (t_1' \, t_2')[t_1/x] : \tau_2$.
This case is proved.

Subcase 2 x $\notin$ dom(Δ_1)

$$\frac{\Delta_1 \vdash_{n_1'} t_1' : \tau_1 \to \tau_2 \ (\star) \qquad \Delta_2, x:[\tau_1]_p \vdash_{n_2'} t_2' : \tau_1 \ (\clubsuit)}{\max(\Delta_1 \Delta_2, x:[\tau_1]_p) \vdash_{\max(n_1', n_2')} t_1' \, t_2' : \tau_2} \text{ APP}$$

This case is proved by induction hypothesis on $(\clubsuit)$ and applying APP rule.

Subcase 3

$$\frac{\Delta_1, x:[\tau_1]_p \vdash_{n_1'} t_1' : \tau_1 \to \tau_2 \ (\star) \qquad \Delta_2, x:[\tau_1]_p \vdash_{n_2'} t_2' : \tau_1 \ (\clubsuit)}{\max(\Delta_1, \Delta_2, x:[\tau_1]_p) \vdash_{\max(n_1', n_2')} t_1' \, t_2' : \tau_2} \text{ APP}$$

To show: $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n_1', n_2'))} t_1' \, t_2' : \tau_2$.
This case is proved by induction hypothesis on both $(\clubsuit)$ and $(\star)$ and then applying APP rule.

Case

$$\frac{\Delta, x:\tau \vdash_n t_2 : \tau_2}{\Delta, x:[\tau]_p \vdash_n t_2 : \tau_2} \text{ DER}$$

Subcase 1

$$\frac{\Delta, x:\tau \vdash_n t_2 : \tau_2}{\Delta, x:[\tau]_p \vdash_n t_2 : \tau_2} \text{ DER}$$

where $n_2 = n$.
To show $\max(\Gamma, \Delta) \vdash_{\max(n_1+p, n)} t_2[t_1/x] : \tau_2$
By induction hypothesis on $(\star)$, this case is proved.

Subcase 2

$$\frac{\Delta', y:\tau, x:[\tau]_p \vdash_n t_2 : \tau_2 \ (\star)}{\Delta', y:[\tau]_{p'}, x:[\tau]_p \vdash_n t_2 : \tau_2} \text{ DER}$$

where $\Delta = \Delta', y:[\tau]_{p'}$ and $n_2 = n$.
To show $\max(\Gamma, \Delta', y:[\tau]_{p'}) \vdash_{\max(n_1+p, n_2)} t_2[t_1/x] : \tau_2$.
By induction hypothesis on $(\star)$, we get: $\max(\Gamma, \Delta', y:\tau) \vdash_{\max(n_1+p, n)} t_2[t_1/x] : \tau_2 \ (\star\star)$.
By applying DER rule on $(\star\star)$, we get: $\max(\Gamma, \Delta', y:[\tau]_p') \vdash_{\max(n_1+p, n)} t_2[t_1/x] : \tau_2$.
This case is proved.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : !_p \tau \qquad \Gamma_2, x:[\tau]_p \vdash_{n_2} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1, n_2)} \mathsf{let}\ !x = t\ \mathsf{in}\ t' : \tau'} \text{ LET}$$

There are three sub cases.

Subcase 1: $\mathbf{x} \not\in \mathbf{dom}(\Delta_2)$

$$\frac{\Delta_1, x:[\tau_1]_p \vdash_{n_1'} t : !_p \tau \;\; (\star) \qquad \Delta_2, y:[\tau]_p \vdash_{n_2'} t' : \tau' \;\; (\clubsuit)}{\max(\Delta_1, x:[\tau_1]_p, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{let}\ !y = t \ \mathsf{in}\ t' : \tau'} \text{ LET}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 =$let $!y = t$ in t', $\tau_2 = \tau'$ and $n_2 = \max(n_1', n_2')$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n_1', n_2'))} (\mathsf{let}\ !y = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By induction hypothesis on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1+p, n_1')} t[t_1/x] : !_p \tau$ ($\star\star$).
$x \not\in \mathrm{dom}(\Delta_2) \implies t'[t_1/x] = t'$.
It is proved by the rule LET using ($\star\star$) and ($\clubsuit$).

Subcase 2: $\mathbf{x} \not\in \mathbf{dom}(\Delta_1)$

$$\frac{\Delta_1 \vdash_{n_1'} t : !_p \tau \;\; (\star) \qquad \Delta_2, x:[\tau_1]_p, y:[\tau]_p \vdash_{n_2'} t' : \tau' \;\; (\clubsuit)}{\max(\Delta_1, x:[\tau_1]_p, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{let}\ !y = t \ \mathsf{in}\ t' : \tau'} \text{ LET}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 =$let $!y = t$ in t', $\tau_2 = \tau'$ and $n_2 = \max(n_1', n_2')$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n_1', n_2'))} (\mathsf{let}\ !y = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By induction hypothesis on ($\clubsuit$), we get $\max(\Gamma, \Delta_2), y:[\tau]_p \vdash_{\max(n_1+p, n_2')} t'[t_1/x] : \tau'$ ($\clubsuit\clubsuit$).
$x \not\in \mathrm{dom}(\Delta_1) \implies t[t_1/x] = t$.
It is proved by the rule LET using ($\star$) and ($\clubsuit\clubsuit$).

Subcase 3

$$\frac{\Delta_1, x:[\tau_1]_p \vdash_{n_1'} t : !_p \tau \;\; (\star) \qquad \Delta_2, x:[\tau_1], y:[\tau]_p \vdash_{n_2'} t' : \tau' \;\; (\clubsuit)}{\max(\Delta_1, x:[\tau_1]_p, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{let}\ !y = t \ \mathsf{in}\ t' : \tau'} \text{ LET}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 =$let $!y = t$ in t', $\tau_2 = \tau'$ and $n_2 = \max(n_1', n_2')$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n_1', n_2'))} (\mathsf{let}\ !y = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By induction hypothesis on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1+p, n_1')} t[t_1/x] : !_p \tau$ ($\star\star$).
By induction hypothesis on ($\clubsuit$), we get $\max(\Gamma, \Delta_2), y:[\tau]_p \vdash_{\max(n_1+p, n_2')} t'[t_1/x] : \tau'$ ($\clubsuit\clubsuit$).
It is proved by the rule LET using ($\star\star$) and ($\clubsuit\clubsuit$).

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : \tau_1 \times \tau_2 \qquad \Gamma_2, x_1 : \tau_1, x_2 : \tau_2 \vdash_{n_2} t' : \tau'}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1, n_2)} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'} \text{ LET-P}$$

Subcase 1: $\mathbf{x} \not\in \mathbf{dom}(\Delta_2)$

$$\frac{\Delta_1, x:[\tau_1]_p \vdash_{n_1'} t : \tau_1' \times \tau_2' \;\; (\star) \qquad \Delta_2, x_1 : \tau_1', x_2 : \tau_2' \vdash_{n_2'} t' : \tau' \;\; (\clubsuit)}{\max(\Delta_1, x:[\tau_1]_p, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'} \text{ LET-P}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 =$let $(x_1, x_2) = t$ in t', $\tau_2 = \tau'$ and $n_2 = \max(n_1', n_2')$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n_1', n_2'))} (\mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t')[t_1/x] : \tau'$.
By induction hypothesis on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1+p, n_1')} t[t_1/x] : \tau_1' \times \tau_2'$ ($\star\star$).
$x \not\in \mathrm{dom}(\Delta_2) \implies t'[t_1/x] = t'$.
It is proved by the rule LET-P using ($\star\star$) and ($\clubsuit$).

Subcase 2: $\mathbf{x} \not\in \mathbf{dom}(\Delta_1)$

$$\frac{\Delta_1 \vdash_{n_1'} t : \tau_1' \times \tau_2' \;\; (\star) \qquad \Delta_2, x_1 : \tau_1', x_2 : \tau_2', x:[\tau_1]_p \vdash_{n_2'} t' : \tau' \;\; (\clubsuit)}{\max(\Delta_1, x:[\tau_1]_p, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{let}\ (x_1, x_2) = t\ \mathsf{in}\ t' : \tau'} \text{ LET-P}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 =$let $(x_1, x_2) = t$ in t', $\tau_2 = \tau'$ and $n_2 = \max(n'_1, n'_2)$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n'_1, n'_2))} (\text{let } (x_1, x_2) = t \text{ in } t')[t_1/x] : \tau'$.
By induction hypothesis on (♣), we get $\max(\Gamma, \Delta_2), x_1 : \tau'_1, x_2 : \tau'_2 \vdash_{\max(n_1+p, n'_2)} t'[t_1/x] : \tau'$ (♣♣).
$x \not\in \mathrm{dom}(\Delta_1) \implies t[t_1/x] = t$.
It is proved by the rule LET-P using ($\star$) and (♣♣).

Subcase 3

$$\frac{\Delta_1, x : [\tau_1]_p \vdash_{n'_1} t : \tau'_1 \times \tau'_2 \ (\star) \qquad \Delta_2, x_1 : \tau'_1, x_2 : \tau'_2, x : [\tau_1]_p \vdash_{n'_2} t' : \tau' \ (\clubsuit)}{\max(\Delta_1, x : [\tau_1]_p, \Delta_2) \vdash_{\max(n'_1, n'_2)} \text{let } (x_1, x_2) = t \text{ in } t' : \tau'} \text{ LET-P}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 =$let $(x_1, x_2) = t$ in t', $\tau_2 = \tau'$ and $n_2 = \max(n'_1, n'_2)$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n'_1, n'_2))} (\text{let } (x_1, x_2) = t \text{ in } t')[t_1/x] : \tau'$.
By induction hypothesis on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1+p, n'_1)} t[t_1/x] : \tau'_1 \times \tau'_2$ ($\star\star$).
By induction hypothesis on (♣), we get $\max(\Gamma, \Delta_2), x_1 : \tau'_1, x_2 : \tau'_2 \vdash_{\max(n_1+p, n'_2)} t'[t_1/x] : \tau'$ (♣♣).
It is proved by the rule LET-P using ($\star\star$) and (♣♣).

Case

$$\frac{\Gamma_1 \vdash_{n_1} t_1 : \tau_1 \qquad \Gamma_2 \vdash_{n_2} t_2 : \tau_2}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1, n_2)} (t_1, t_2) : \tau_1 \times \tau_2} \text{ PAIR}$$

There are three sub cases,x only in Γ_1, x only in Γ_2 and x appears in both Γ_1 and Γ_2. When x only in Γ_1, it is proved by induction hypothesis on the first premise and then using the rule PAIR. When x only in Γ_2, it is proved by induction hypothesis on the second premise and then using the rule PAIR. When x appears in both Γ_1 and Γ_2, it is proved by induction hypothesis on both premises and then using rule PAIR.

Case

$$\frac{\Gamma_1 \vdash_{n_1} t : b \qquad \Gamma_2 \vdash_{n_2} t_i : b}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1, n_2)} \text{case } t \text{ of } \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \text{ CASE-CONST}$$

There are three sub cases.

Subcase 1: x $\not\in$ dom(Δ_2)

$$\frac{\Delta_1, x : [\tau_1]_p \vdash_{n'_1} t : b \ (\star) \qquad \Delta_2 \vdash_{n'_2} t_i : b \ (\clubsuit)}{\max(\Delta_1, x : [\tau_1]_p, \Delta_2) \vdash_{\max(n'_1, n'_2)} \text{case } t \text{ of } \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \text{ CASE-CONST}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 = \text{case } t \text{ of } \{c_i \Rightarrow t_i\}_{c_i \in b}$, $\tau_2 = b$ and $n_2 = \max(n'_1, n'_2)$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n'_1, n'_2))} (\text{case } t \text{ of } \{c_i \Rightarrow t_i\}_{c_i \in b})[t_1/x] : b$.
By induction hypothesis on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1+p, n'_1)} t[t_1/x] : b$ ($\star\star$).
$x \not\in \mathrm{dom}(\Delta_2) \implies t_i[t_1/x] = t_i$.
It is proved by the rule CASE-CONST using ($\star\star$) and (♣).

Subcase 2: x $\not\in$ dom(Δ_1)

$$\frac{\Delta_1 \vdash_{n'_1} t : b \ (\star) \qquad \Delta_2, x : [\tau_1]_p \vdash_{n'_2} t_i : b \ (\clubsuit)}{\max(\Delta_1, x : [\tau_1]_p, \Delta_2) \vdash_{\max(n'_1, n'_2)} \text{case } t \text{ of } \{c_i \Rightarrow t_i\}_{c_i \in b} : b} \text{ CASE-CONST}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 = \text{case } t \text{ of } \{c_i \Rightarrow t_i\}_{c_i \in b}$, $\tau_2 = b$ and $n_2 = \max(n'_1, n'_2)$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n'_1, n'_2))} (\text{case } t \text{ of } \{c_i \Rightarrow t_i\}_{c_i \in b})[t_1/x] : b$.
By induction hypothesis on (♣), we get $\max(\Gamma, \Delta_2) \vdash_{\max(n_1+p, n'_2)} t_i[t_1/x] : b$ (♣♣).
$x \not\in \mathrm{dom}(\Delta_1) \implies t[t_1/x] = t$.

It is proved by the rule CASE-CONST using ($\star$) and ($\clubsuit\clubsuit$).

Subcase 3

$$\frac{\Delta_1, x:[\tau_1]_p \vdash_{n_1'} t:b \ (\star) \qquad \Delta_2, x:[\tau_1]_p \vdash_{n_2'} t_i:b \ (\clubsuit)}{\max(\Delta_1, x:[\tau_1]_p, \Delta_2) \vdash_{\max(n_1', n_2')} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : b}\ \text{CASE-CONST}$$

where $\Delta = \max(\Delta_1, \Delta_2)$, $t_2 = \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b}$, $\tau_2 = b$ and $n_2 = \max(n_1', n_2')$.
To show $\max(\Gamma, \max(\Delta_1, \Delta_2)) \vdash_{\max(n_1+p, \max(n_1', n_2'))} (\mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b})[t_1/x] : b$.
By induction hypothesis on ($\star$), we get $\max(\Gamma, \Delta_1) \vdash_{\max(n_1+p, n_1')} t[t_1/x] : b$ ($\star\star$).
By induction hypothesis on ($\clubsuit$), we get $\max(\Gamma, \Delta_2) \vdash_{\max(n_1+p, n_2')} t_i[t_1/x] : b$ ($\clubsuit\clubsuit$)..
It is proved by the rule CASE-CONST using ($\star\star$) and ($\clubsuit\clubsuit$).

Case

$$\frac{\Gamma_1 \vdash_{n_1'} t:b \qquad \Gamma_2 \vdash_{n_2'} t_i : query}{\max(\Gamma_1, \Gamma_2) \vdash_{\max(n_1', n_2')} \mathsf{case}\ t\ \mathsf{of}\ \{c_i \Rightarrow t_i\}_{c_i \in b} : query}\ \text{CASE-QUERY}$$

There are three sub cases. Thre proof are quite similar as the one of CASE-CONST.

Case

$$\frac{i :: \mathbb{N}; \Delta, x:[\tau_1]_p \vdash_n t:\tau \ (\star) \qquad i \notin \mathrm{FIV}(\Delta, x:\tau_1)}{\Delta, x:[\tau_1]_p \vdash_n \Lambda.t : \forall i{::}\mathbb{N}.\tau}\ \text{IABS}$$

To show $\max(\Gamma, \Delta) \vdash_{\max(n_1+p, n)} \Lambda.t[t_1/x] : \forall i{::}\mathbb{N}.\tau$.
By induction hypothesis on ($\star$), we get : $i :: \mathbb{N}; \max(\Gamma, \Delta) \vdash_{\max(n_1, n)} t[t_1/x] : \tau$ ($\star\star$).
There are two cases.
Sub case 1: $i \notin \mathbf{FIV}(\Gamma)$ It is proved by using rule IABS with ($\star\star$).
Sub case 2: $i \in \mathbf{FIV}(\Gamma)$ We choose $j \notin \mathrm{FIV}(\max(\Gamma, \Delta))$. We rename all the i to j in ($\star\star$) and use rule IABS to get: $\max(\Gamma, \Delta) \vdash_{\max(n_1, n)} \Lambda.t[t_1/x] : \forall j{::}\mathbb{N}.\tau$. It is just a renaming version of the goal.
Case

$$\frac{\Delta, x:[\tau_1]_p \vdash_n t : \forall i{::}\mathbb{N}.\tau \ (\star) \qquad \vdash I :: \mathbb{N}}{\Delta, x:[\tau_1]_p \vdash_n t\,[\,] : \tau\{I/i\}}\ \text{IAPP}$$

To show $\max(\Gamma, \Delta) \vdash_{\max(n_1+p, n)} t[t_1/x]\,[\,] : \tau\{I/i\}$.
By ih on ($\star$), we get: $\max(\Gamma, \Delta) \vdash_{\max(n_1+p, n)} t : \forall i{::}\mathbb{N}.\tau$ ($\star\star$).
It is proved by the rule IAPP with ($\star\star$).

Case

$$\frac{\Delta, x:[\tau_1]_p \vdash_n t:\tau \ (\star) \qquad \Delta', x:[\tau_1]_p \sqsubseteq \Delta, x:\tau_1 \ (\clubsuit) \qquad \vDash n \leq n' \ (\spadesuit) \qquad \tau \sqsubseteq \tau'}{\Delta', x:[\tau_1]_p \vdash_{n'} t:\tau'}\ \text{SUB}$$

To show $\max(\Gamma, \Delta') \vdash_{\max(n_1+p, n')} t : \tau'$.
By induction hypothesis on ($\star$), we get : $\max(\Gamma, \Delta) \vdash_{\max(n_1+p, n)} t : \tau$.
($\clubsuit$) $\implies \max(\Gamma, \Delta') \sqsubseteq \max(\Gamma, \Delta)$ ($\clubsuit\clubsuit$).
($\spadesuit$) $\implies \vDash \max(n_1+p, n) \leq \max(n_1+p, n')$.
It is proved by the rule SUB.

Case

$$\frac{[\Delta_2], x:[\tau_1]_{p-p'} \vdash_n t:\tau}{\Delta_1, p' + [\Delta_2], x:[\tau_1]_p \vdash_{n+p'} !t : !'_p\tau}\ \text{PR}$$

TS: $\max(\Gamma, (\Delta_1, p + [\Delta_2])) \vdash_{\max(n_1+p, n+p')} !t[t_1/x] : !_{p'}\tau$.
By induction hypothesis on premise, we get: .

By rule PR with ($\star$), we get: $\Delta_1, p' + [\Delta_2] \vdash_{n+p'} !t[t_1/x] : !_{p'}\tau$. It is proved by the Lemma 7 and Lemma 3 from the conclusion.

$\square$