1 Mutable references [50 pts]

Here is the definition of the simply-typed λ-calculus extended with mutable references:

\[
\begin{align*}
\text{type} & \quad A ::= P \mid A \to A \mid \text{unit} \mid \text{ref } A \\
\text{expression} & \quad e ::= x \mid \lambda x : A. e \mid e \ e \mid () \mid \text{ref } e \mid !e \mid e ::= e \mid l \\
\text{value} & \quad v ::= \lambda x : A. e \mid () \mid l \\
\text{store} & \quad \psi ::= \cdot \mid \psi, l \mapsto v \\
\text{store typing context} & \quad \Psi ::= \cdot \mid \Psi, l \mapsto A
\end{align*}
\]

We use the following reduction judgment and typing judgment:

\[
e | \psi \mapsto e' \mid \psi' \iff e \text{ with store } \psi \text{ reduces to } e' \text{ with store } \psi'
\]

\[
\Gamma | \Psi \vdash e : A \iff \text{expression } e \text{ has type } A \text{ under typing context } \Gamma \text{ and store typing context } \Psi
\]

Write the typing rule for location \( l \):

\[
\frac{\Psi(l) = A}{\Gamma | \Psi \vdash l : \text{ref } A} \text{ Loc}
\]

Write a rule for the judgment \( \psi :: \Psi \) which means that \( \psi \) is well-typed with \( \Psi \):

\[
\frac{\text{dom}(\Psi) = \text{dom}(\psi) \cdot | \Psi \vdash \psi(l) : \Psi(l) \text{ for every } l \in \text{dom}(\psi)}{\psi :: \Psi} \text{ Store}
\]
2 Subtyping [50 pts]

Give subtyping rules for product types, function types, and reference types:

\[
\begin{align*}
A \leq A' & \quad B \leq B' \\
A \times B & \leq A' \times B' & \text{Prod}_\leq \\
A' \leq A & \quad B \leq B' \\
A \to B & \leq A' \to B' & \text{Fun}_\leq \\
A \leq B & \quad B \leq A \\
\text{ref } A & \leq \text{ref } B & \text{Ref}_\leq
\end{align*}
\]