1 Subtyping [50 pts]

Consider the following definition:

\[
\text{type} \quad A ::= \quad P \mid A \to A \mid A \times A \mid A + A \mid \text{ref } A \\
\text{base type} \quad P ::= \quad \text{nat} \mid \text{int}
\]

We write \( A \leq B \) if \( A \) is a subtype of \( B \), or equivalently, if \( B \) is a supertype of \( A \). Complete the following subtyping rules for product types, sum types and function types:

\[
A \times B \leq A' \times B'
\]

\[
A + B \leq A' + B'
\]

\[
A \to B \leq A' \to B'
\]
2 Recursive Types [50 pts]

Consider the following definition of simply typed λ-calculus extended with recursive types:

\[
\begin{align*}
\text{type} & \quad A ::= \text{unit} \mid A \to A \mid A + A \mid \alpha \mid \mu \alpha.A \\
\text{expression} & \quad e ::= x \mid \lambda x:A.e \mid e e \mid () \mid \text{inl}_A e \mid \text{inr}_A e \mid \text{case } e \text{ of } \text{inl } x.e \mid \text{inr } y.e \mid \text{fold}_C e \mid \text{unfold}_C e \\
\text{typing context} & \quad \Gamma ::= \cdot \mid \Gamma, x:A \mid \Gamma, \alpha \text{ type}
\end{align*}
\]

Question 1. [20 pts] Complete the typing rules for \(\text{fold}_C e\) and \(\text{unfold}_C e\):

\[
\Gamma \vdash \text{fold}_C e : C
\]

\[
\Gamma \vdash \text{unfold}_C e : [C/\alpha]A
\]

Question 2. [30 pts] Complete the following reduction rules based on the eager reduction strategy:

\[
\text{fold}_C e \mapsto
\]

\[
\text{unfold}_C e \mapsto
\]

\[
\text{unfold}_C \text{ fold}_C v \mapsto
\]