CS 565 Spring 2022 Homework 6
(Type Inference + Subtyping)

Your name: Shuang Wu (wu1716@purdue.edu)

April 27, 2022

Problem 1 (1 point). Construct a constraint typing derivation whose conclusion is

$$\vdash \lambda \mathsf{x} : \mathsf{X}.\ \lambda \mathsf{y} : \mathsf{Y}.\ \lambda \mathsf{z} : \mathsf{Z}.\ (\mathsf{x}\ \mathsf{z})\ (\mathsf{y}\ \mathsf{z}) : \mathsf{S} \ \mid\ \mathcal{C}$$

for some S, $\mathcal{C}$.

$$
\dfrac{
 \dfrac{
  \dfrac{
   \dfrac{
    \dfrac{
     \text{CTVar}\ \dfrac{[x \mapsto X, y \mapsto Y, z \mapsto Z](x) : X}{[x \mapsto X, y \mapsto Y, z \mapsto Z] \vdash x : X \ \mid\ \emptyset}
     \qquad
     \dfrac{[x \mapsto X, y \mapsto Y, z \mapsto Z](z) : Z}{[x \mapsto X, y \mapsto Y, z \mapsto Z] \vdash z : Z \ \mid\ \emptyset}\ \text{CTVar}
    }{[x \mapsto X, y \mapsto Y, z \mapsto Z] \vdash (x\ z) : T_{xz} \ \mid\ \{X = Z \to T_{xz}\}}\ \text{CTApp}
    \qquad
    \dfrac{
     \text{CTVar}\ \dfrac{[x \mapsto X, y \mapsto Y, z \mapsto Z](y) : Y}{[x \mapsto X, y \mapsto Y, z \mapsto Z] \vdash y : Y \ \mid\ \emptyset}
     \qquad
     \dfrac{[x \mapsto X, y \mapsto Y, z \mapsto Z](z) : Z}{[x \mapsto X, y \mapsto Y, z \mapsto Z] \vdash z : Z \ \mid\ \emptyset}\ \text{CTVar}
    }{[x \mapsto X, y \mapsto Y, z \mapsto Z] \vdash (y\ z) : T_{yz} \ \mid\ \{Y = Z \to T_{yz}\}}\ \text{CTApp}
   }{[x \mapsto X, y \mapsto Y, z \mapsto Z] \vdash (x\ z)\ (y\ z) : T \ \mid\ \mathcal{C}}\ \text{CTApp}
  }{[x \mapsto X, y \mapsto Y] \vdash \lambda z : Z.\ (x\ z)\ (y\ z) : Z \to T \ \mid\ \mathcal{C}}\ \text{CTAbs}
 }{[x \mapsto X] \vdash \lambda y : Y.\ \lambda z : Z.\ (x\ z)\ (y\ z) : Y \to Z \to T \ \mid\ \mathcal{C}}\ \text{CTAbs}
}{\vdash \lambda x : X.\ \lambda y : Y.\ \lambda z : Z.\ (x\ z)\ (y\ z) : S \ \mid\ \mathcal{C}}\ \text{CTAbs}
$$

$\mathcal{C} \equiv \{X = Z \to T_{xz}, Y = Z \to T_{yz}, T_{xz} = T_{yz} \to T\}$
$S \equiv X \to Y \to Z \to T$

Problem 2 (2 points). Write down principal unifers (when they exist) for the following sets of constraints:

- $\{\}$ (The empty set of constraints) $\equiv \{\}$
- $\{\mathsf{Y} = \mathsf{V} \to \mathsf{U}, \mathsf{Y} = \mathsf{X} \to \mathsf{V}\} \equiv \{\mathsf{X} \to \mathsf{X} = \mathsf{X} \to \mathsf{X}, \mathsf{X} \to \mathsf{X} = \mathsf{X} \to \mathsf{X}\}$
- $\{\mathsf{X} = \mathsf{Bool}, \mathsf{Y} = \mathsf{X} \to \mathsf{X}\} \equiv \{\mathsf{Bool} = \mathsf{Bool}, \mathsf{Bool} \to \mathsf{Bool} = \mathsf{Bool} \to \mathsf{Bool}\}$
- $\{\mathsf{Bool} \to \mathsf{Bool} = \mathsf{X} \to \mathsf{Y}\} \equiv \{\mathsf{Bool} \to \mathsf{Bool} = \mathsf{Bool} \to \mathsf{Bool}\}$
- $\{(\mathsf{Bool} \to \mathsf{Y}) \to \mathsf{Bool} = \mathsf{Bool} \to \mathsf{U}\}$ fail to unify.

Problem 3 (2 points). Suppose we have types S, T, U, and V with $\mathsf{S} <: \mathsf{T}$ and $\mathsf{U} <: \mathsf{V}$. Which of the following subtyping assertions are then true? Write true or false after each one.

- $\mathsf{T}\to\mathsf{S} <: \mathsf{T}\to\mathsf{S}$ True.
- $\mathsf{T}\to\mathsf{T}\to\mathsf{U} <: \mathsf{S}\to\mathsf{S}\to\mathsf{V}$ True.
- $(\mathsf{T}\to\mathsf{T})\to\mathsf{U} <: (\mathsf{S}\to\mathsf{S})\to\mathsf{V}$ False.
- $((\mathsf{T}\to\mathsf{S})\to\mathsf{T})\to\mathsf{U} <: ((\mathsf{S}\to\mathsf{T})\to\mathsf{S})\to\mathsf{V}$ True.

Problem 4 (1 point). How many supertypes does the type

$$\{\{\mathsf{x}\text{: }\{\mathsf{z}\text{:}\mathsf{Bool},\ \mathsf{q}\text{: }\mathsf{Nat}\},\ \mathsf{y}\text{: }\mathsf{Bool} \to \mathsf{Bool}\}\}$$

have? That is, how many different types T are there such that

$$\{\mathsf{x}\text{: }\{\mathsf{z}\text{:}\mathsf{Bool},\ \mathsf{q}\text{: }\mathsf{Nat}\},\ \mathsf{y}\text{: }\mathsf{Bool} \to \mathsf{Bool}\} <: \mathsf{T}$$

(We consider two types to be different if they are written differently, even if each is a subtype of the other. For example, $\{\mathsf{x}\text{:}\mathsf{A},\mathsf{y}\text{:}\mathsf{B}\}$ and $\{\mathsf{y}\text{:}\mathsf{B},\mathsf{x}\text{:}\mathsf{A}\}$ are different.)

The record has 17 supertypes:
$\{\}$,
$\{y : \text{Bool} \to \text{Bool}\}$,
$\{x : \{\}\}$,
$\{x : \{z : \text{Bool}\}\}$,
$\{x : \{q : \text{Nat}\}\}$,
$\{x : \{z : \text{Bool}, q : \text{Nat}\}\}$,
$\{x : \{q : \text{Nat}, z : \text{Bool}\}\}$,
$\{x : \{\}, y : \text{Bool} \to \text{Bool}\}$,
$\{x : \{z : \text{Bool}\}, y : \text{Bool} \to \text{Bool}\}$,
$\{x : \{q : \text{Nat}\}, y : \text{Bool} \to \text{Bool}\}$,
$\{x : \{z : \text{Bool}, q : \text{Nat}\}, y : \text{Bool} \to \text{Bool}\}$,
$\{x : \{q : \text{Nat}, z : \text{Bool}\}, y : \text{Bool} \to \text{Bool}\}$
$\{y : \text{Bool} \to \text{Bool}, x : \{\}\}$,
$\{y : \text{Bool} \to \text{Bool}, x : \{z : \text{Bool}\}\}$,
$\{y : \text{Bool} \to \text{Bool}, x : \{q : \text{Nat}\}\}$,
$\{y : \text{Bool} \to \text{Bool}, x : \{z : \text{Bool}, q : \text{Nat}\}\}$,
$\{y : \text{Bool} \to \text{Bool}, x : \{q : \text{Nat}, z : \text{Bool}\}\}$

Problem 5 (2 points). The subtyping rule for product types:

$$\frac{\mathsf{S}_1 <: \mathsf{T}_1 \qquad \mathsf{S}_2 <: \mathsf{T}_2}{\mathsf{S}_1 * \mathsf{S}_2 <: \mathsf{T}_1 * \mathsf{T}_2}$$

intuitively corresponds to the "depth" subtyping rule for records. Extending the analogy, a language designer might consider including a "permutation" rule as well

$$\frac{}{\mathsf{T}_1 * \mathsf{T}_2 <: \mathsf{T}_2 * \mathsf{T}_1}$$

for products. Explain in a couple of sentences why such a subtyping rule is or is not sound?

This "permutation" rule is not sound. For instance, assume $T_1 \equiv \text{Nat}$ and $T_2 \equiv \text{Bool}$, $\text{Nat} * \text{Bool} <: \text{Bool} * \text{Nat}$ means any value of type $\text{Nat} * \text{Bool}$ must be usable in every way a $\text{Bool}*\text{Nat}$ is. However, this does not hold for all values. Consider the case that $v_1 = (1, \text{true})$ and $v_2 = (\text{true}, 1)$. v_2 is usable in

$$\textbf{if } \text{fst } v_2 \textbf{ then } \text{true} \textbf{ else } \text{false}$$

but v_1 is not.