Ungraded Problem Set #5

1. Extend the *constraint typing rules* presented on page 2 to support a new constraint typing rule corresponding to (T-Rcd) from page 3.

2. Define a *constraint typing rule* for the modified version of (T-Proj) given below.

$$\frac{\Gamma \vdash t_1 : \{l_i : \tau_i \;^{i \in 1..n}\}}{\Gamma \vdash (t_1 \text{ with } \{l_i \;^{i \in 1..n}\}).j : \tau_j} \quad \text{(T-ProjAsc)}$$

3. Using the *constraint typing rules* presented on page 2 and your extensions developed in the previous problems, give a type derivation to compute the constraints for each of the following assuming the specified type context (Γ) to start. Complete the derivation even if the constraints cannot be met; you are not expected to solve the constraints.

   - (λr. r) {x = 2, y = 3}
     where $\Gamma = \{\}$
   - (λr. (r with {x:Int}).x) a
     where $\Gamma = \{$a : {x:Int, y:Int}$\}$
   - λf. λg. g (f g)
     where $\Gamma = \{\}$

4. Using the typing rules (page 3) with the rules for polymorphism (page 4), give a type derivation for each of the following assuming the specified type context (Γ) to start.

   - λX. λY. λf: (X $\rightarrow$ Y) $\rightarrow$ X. λg: X $\rightarrow$ Y. g (f g)
     where $\Gamma = \{\}$
   - λf : $\forall$X.X $\rightarrow$ Int. (f [Int] a) + (f [Bool] b)
     where Γ = {a:Int, b:Bool}
   - λX.λf : X $\rightarrow$ Int. (f a) + (f b)
     where Γ = {a:Int, b:Bool}

5. Consider the following "counter" represented as an existential type. Give an alternate implementation using a record with the same existential type.

```
counter =  {*Nat,
              {
              value = 0,
              increment = λv : Nat. succ(v),
              get = λv : Nat. v
              }
           }
as         {∃X,
              {
              value : X,
              increment : X → X,
              get : X → Nat
              }
           }
```

The terms for the expression language are to be inferred from the rules below coupled with the discussion in lecture (and in the textbook).

nv is for numeric values.

Constraint Typing Rules

$$\Gamma \vdash \text{true} : \text{Bool} \mid \{\} \tag{CT-True}$$

$$\Gamma \vdash \text{false} : \text{Bool} \mid \{\} \tag{CT-False}$$

$$\frac{\Gamma \vdash t_1 : \tau_1 \mid C_1 \quad \Gamma \vdash t_2 : \tau_2 \mid C_2 \quad \Gamma \vdash t_3 : \tau_3 \mid C_3 \quad C' = C_1 \cup C_2 \cup C_3 \cup \{\tau_1 = \text{Bool}, \tau_2 = \tau_3\}}{\Gamma \vdash \text{if } t_1 \text{ then } t_2 \text{ else } t_3 : \tau_2 \mid C'} \tag{CT-If}$$

$$\Gamma \vdash nv : \text{Int} \mid \{\} \tag{CT-IntConst}$$

$$\frac{t_1 : \tau_1 \mid C_1 \quad C' = C_1 \cup \{\tau_1 = \text{Int}\}}{\Gamma \vdash \text{iszero } t_1 : \text{Bool} \mid C'} \tag{CT-IsZero}$$

$$\frac{\Gamma \vdash t_1 : \tau_1 \mid C_1 \quad \Gamma \vdash t_2 : \tau_2 \mid C_2 \quad C' = C_1 \cup C_2 \cup \{\tau_1 = \text{Int}, \tau_2 = \text{Int}\}}{\Gamma \vdash t_1 + t_2 : \text{Int} \mid C'} \tag{CT-Add}$$

$$\frac{\Gamma \vdash t_1 : \tau_1 \mid C_1 \quad \Gamma \vdash t_2 : \tau_2 \mid C_2 \quad C' = C_1 \cup C_2 \cup \{\tau_1 = \text{Int}, \tau_2 = \text{Int}\}}{\Gamma \vdash t_1 - t_2 : \text{Int} \mid C'} \tag{CT-Sub}$$

$$\frac{\text{x} : \tau \in \Gamma}{\Gamma \vdash \text{x} : \tau \mid \{\}} \tag{CT-Var}$$

$$\frac{\Gamma, \text{x} : \alpha \vdash t_1 : \beta \mid C}{\Gamma \vdash \lambda \text{x} : \alpha.\ t_1 : \alpha \to \beta \mid C} \tag{CT-Abs}$$

$$\frac{\Gamma \vdash t_1 : \tau_1 \mid C_1 \quad \Gamma \vdash t_2 : \tau_2 \mid C_2 \quad C' = C_1 \cup C_2 \cup \{\tau_1 = \tau_2 \to \text{X}\}}{\Gamma \vdash t_1\ t_2 : \text{X}} \tag{CT-App}$$

where X is a fresh variable

Typing Rules

$$\Gamma \vdash \text{true} : \text{Bool} \qquad \text{(T-True)}$$

$$\Gamma \vdash \text{false} : \text{Bool} \qquad \text{(T-False)}$$

$$\frac{\Gamma \vdash t_1 : \text{Bool} \quad \Gamma \vdash t_2 : \tau \quad \Gamma \vdash t_3 : \tau}{\Gamma \vdash \text{if } t_1 \text{ then } t_2 \text{ else } t_3 : \tau} \qquad \text{(T-If)}$$

$$\Gamma \vdash nv : \text{Int} \qquad \text{(T-IntConst)}$$

$$\frac{t_1 : Int}{\Gamma \vdash \text{iszero } t_1 : \text{Bool}} \qquad \text{(T-IsZero)}$$

$$\frac{\Gamma \vdash t_1 : \text{Int} \quad \Gamma \vdash t_2 : \text{Int}}{\Gamma \vdash t_1 + t_2 : \text{Int}} \qquad \text{(T-Add)}$$

$$\frac{\Gamma \vdash t_1 : \text{Int} \quad \Gamma \vdash t_2 : \text{Int}}{\Gamma \vdash t_1 - t_2 : \text{Int}} \qquad \text{(T-Sub)}$$

$$\frac{\text{x} : \tau \in \Gamma}{\Gamma \vdash \text{x} : \tau} \qquad \text{(T-Var)}$$

$$\frac{\Gamma, \text{x} : \alpha \vdash t_1 : \beta}{\Gamma \vdash \lambda \text{x} : \alpha.\ t_1 : \alpha \to \beta} \qquad \text{(T-Abs)}$$

$$\frac{\Gamma \vdash t_1 : \alpha \to \beta \quad \Gamma \vdash t_2 : \alpha}{\Gamma \vdash t_1\ t_2 : \beta} \qquad \text{(T-App)}$$

$$\frac{\text{for each } i \quad \Gamma \vdash t_i : \tau_i}{\Gamma \vdash \{l_i = t_i\ ^{i \in 1..n}\} : \{l_i : \tau_i\ ^{i \in 1..n}\}} \qquad \text{(T-Rcd)}$$

$$\frac{\Gamma \vdash t_1 : \{l_i : \tau_i\ ^{i \in 1..n}\}}{\Gamma \vdash t_1.j : \tau_j} \qquad \text{(T-Proj)}$$

$$\frac{\Gamma \vdash t_j : \tau_j}{\Gamma \vdash \langle l_j = t_j \rangle \text{ as } \langle l_i : \tau_i\ ^{i \in 1..n} \rangle : \langle l_i : \tau_i\ ^{i \in 1..n} \rangle} \qquad \text{(T-Variant)}$$

$$\frac{\Gamma \vdash t_0 : \langle l_i : \tau_i\ ^{i \in 1..n} \rangle \quad \text{for each } i \quad \Gamma, x_i : \tau_i \vdash t_i : \tau}{\Gamma \vdash \text{case } t_0 \text{ of } \langle l_i = x_i \rangle \Rightarrow t_i\ ^{i \in 1..n} : \tau} \qquad \text{(T-Case)}$$

$$\Gamma \vdash \text{unit} : \text{Unit} \qquad \text{(T-Unit)}$$

Polymorphism Rules

$$\frac{\Gamma, \mathrm{X} \vdash t_1 : \tau}{\Gamma \vdash \lambda \mathrm{X}.\ t_1 : \forall X.\tau} \qquad \text{(T-TAbs)}$$

$$\frac{\Gamma \vdash t_1 : \forall \mathrm{X}.\tau}{\Gamma \vdash t_1\ [\beta] : [\mathrm{X} \mapsto \beta]\tau} \qquad \text{(T-TApp)}$$

$$\frac{\Gamma \vdash t_1 : [\mathrm{X} \mapsto \mathrm{U}]\tau_1}{\Gamma \vdash \{*\mathrm{U}, t_1\} \text{ as } \{\exists \mathrm{X}, \tau_1\} : \{\exists \mathrm{X}, \tau_1\}} \qquad \text{(T-Pack)}$$

$$\frac{\Gamma \vdash t_1 : \{\exists \mathrm{X}, \tau_1\} \quad \Gamma, \mathrm{X}, \mathrm{x} : \tau_1 \vdash t_2 : \tau_2}{\Gamma \vdash \text{ let } \{\mathrm{X}, \mathrm{x}\} = t_1 \text{ in } t_2 : \tau_2} \qquad \text{(T-UnPack)}$$

Subtyping Rules

$$\mathrm{S} <: \mathrm{S} \qquad \text{(S-Refl)}$$

$$\frac{\mathrm{S} <: \mathrm{U} \quad \mathrm{U} <: \mathrm{T}}{\mathrm{S} <: \mathrm{T}} \qquad \text{(S-Trans)}$$

$$\frac{\Gamma \vdash \mathrm{t} : \mathrm{S} \quad \mathrm{S} <: \mathrm{T}}{\Gamma \vdash \mathrm{t} : \mathrm{T}} \qquad \text{(S-Sub)}$$

$$\frac{\mathrm{P}_2 <: \mathrm{P}_1 \quad \mathrm{R}_1 <: \mathrm{R}_2}{\mathrm{P}_1 \to \mathrm{R}_1 <: \mathrm{P}_2 \to \mathrm{R}_2} \qquad \text{(S-Arrow)}$$

$$\frac{\{l_i\ ^{i \in 1..n}\} \subseteq \{k_j\ ^{j \in 1..m}\} \quad k_j = l_i \text{ implies } \alpha_j <: \beta_i}{\{k_j : \alpha_j\ ^{j \in 1..m}\} <: \{l_i : \beta_i\ ^{i \in 1..n}\}} \qquad \text{(S-Rcd)}$$