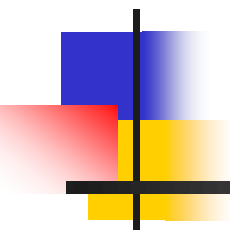


# Programming Languages and Compilers (CS 421)

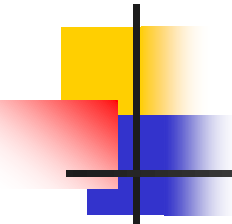


---

William Mansky

<http://courses.engr.illinois.edu/cs421/>

Based in part on slides by Mattox Beckman, as updated by Vikram Adve, Gul Agha, Elsa Gunter, and Dennis Griffith



# Two Problems

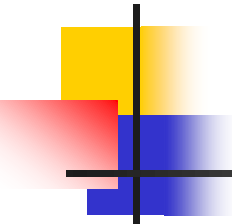
---

- Type checking

- Question: Does exp.  $e$  have type  $\tau$  in env  $\Gamma$ ?
- Answer: Yes / No
- Method: Type **derivation**

- Typability

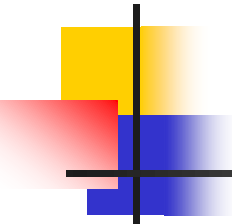
- Question Does exp.  $e$  have **some type** in env  $\Gamma$ ?  
If so, what is it?
- Answer: Type  $\tau$  / error
- Method: Type **inference**



# Type Inference - Outline

---

- Assign type variables to unknown types
- Decompose expressions with typing rules
- Store constraints on relationship between variables
- Find substitution that solves typing judgment
- Use substitution in generating constraints for next branch; repeat
- Apply composition of all substitutions to top-level type variable to get answer



# Type Inference - Example

---

- What type can we give to

`(fun x -> fun f -> f (f x))`

- Start with a type variable and then look at the way the expression is constructed

# Type Inference - Example

- First approximation:

$$[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha$$

- Second approximation: use fun rule

$$\frac{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- Remember constraint  $\alpha \equiv (\beta \rightarrow \gamma)$

# Type Inference - Example

- Third approximation: use fun rule

$$\frac{\frac{[f : \delta; x : \beta] \vdash f (f x) : \varepsilon}{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}}{[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Fourth approximation: use app rule

$$\frac{[f: \delta; x: \beta] \vdash f : \varphi \rightarrow \varepsilon \quad [f: \delta; x: \beta] \vdash f x : \varphi}{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}$$

$$\frac{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}$$

$$\frac{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$
- Fifth approximation: use var rule, get constraint  $\delta \equiv \varphi \rightarrow \varepsilon$

# Type Inference - Example

- Fifth approximation: use var rule, get constraint  $\delta \equiv \varphi \rightarrow \varepsilon$ , solve with same
- Apply to next sub-proof

$$\frac{[f: \delta; x: \beta] \vdash f : \varphi \rightarrow \varepsilon \quad [f: \delta; x: \beta] \vdash f x : \varphi}{[f: \delta; x: \beta] \vdash (f (f x)) : \varepsilon}$$

$$\frac{[f: \delta; x: \beta] \vdash (f (f x)) : \varepsilon}{[x: \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}$$

$$\frac{[x: \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\delta \equiv \varphi \rightarrow \varepsilon\}$

$$\frac{\dots \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f \ x : \varphi}{[f : \delta; x : \beta] \vdash (f \ (f \ x)) : \varepsilon}$$
$$\frac{[x : \beta] \vdash (\text{fun } f \text{ -> } f \ (f \ x)) : \gamma}{[ ] \vdash (\text{fun } x \text{ -> fun } f \text{ -> } f \ (f \ x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\delta \equiv \varphi \rightarrow \varepsilon\}$

$$\frac{[f:\varphi \rightarrow \varepsilon; x:\beta] \vdash f:\zeta \rightarrow \varphi \quad [f:\varphi \rightarrow \varepsilon; x:\beta] \vdash x:\zeta}{\dots \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f x : \varphi}$$
$$\frac{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}{[x : \beta] \vdash (\text{fun } f \text{ -> } f (f x)) : \gamma}$$
$$[ ] \vdash (\text{fun } x \text{ -> fun } f \text{ -> } f (f x)) : \alpha$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\delta \equiv \varphi \rightarrow \varepsilon\}$
- Var rule: Solve  $\zeta \rightarrow \varphi \equiv \varphi \rightarrow \varepsilon$  **Unification**

$$\frac{[f:\varphi \rightarrow \varepsilon; x:\beta] \vdash f:\zeta \rightarrow \varphi \quad [f:\varphi \rightarrow \varepsilon; x:\beta] \vdash x:\zeta}{\dots}$$

$$\frac{\dots \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f x : \varphi}{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}$$

$$\frac{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}{[x : \beta] \vdash (\text{fun } f \text{ -> } f (f x)) : \gamma}$$

$$\frac{[x : \beta] \vdash (\text{fun } f \text{ -> } f (f x)) : \gamma}{[ ] \vdash (\text{fun } x \text{ -> fun } f \text{ -> } f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon\} \circ \{\delta \equiv \varphi \rightarrow \varepsilon\}$
- Var rule: Solve  $\zeta \rightarrow \varphi \equiv \varphi \rightarrow \varepsilon$  **Unification**

$$\frac{[f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f : \zeta \rightarrow \varphi \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash x : \zeta}{\dots}$$

$$\frac{\dots \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f \ x : \varphi}{[f : \delta; x : \beta] \vdash (f \ (f \ x)) : \varepsilon}$$

$$\frac{[x : \beta] \vdash (\text{fun } f \rightarrow f \ (f \ x)) : \gamma}{[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f \ (f \ x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon, \delta \equiv \varepsilon \rightarrow \varepsilon\}$
- Apply to next sub-proof

$$\frac{\dots \quad [f : \varepsilon \rightarrow \varepsilon; x : \beta] \vdash x : \varepsilon}{\dots \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f x : \varphi}$$
$$\frac{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}$$
$$[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon, \delta \equiv \varepsilon \rightarrow \varepsilon\}$

- Var rule:  $\varepsilon \equiv \beta$

$$\frac{\dots \quad \frac{[f : \varepsilon \rightarrow \varepsilon; x : \beta] \vdash x : \varepsilon}{[f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f x : \varphi}}{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}$$
$$\frac{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\varepsilon \equiv \beta\} \circ \{\zeta \equiv \varepsilon, \varphi \equiv \varepsilon, \delta \equiv \varepsilon \rightarrow \varepsilon\}$
- Solves subproof; return one layer

$$\begin{array}{c}
 \dots \quad \overline{[f : \varepsilon \rightarrow \varepsilon; x : \beta] \vdash x : \varepsilon} \\
 \hline
 \dots \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f x : \varphi \\
 \hline
 [f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon \\
 \hline
 [x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma \\
 \hline
 [ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha
 \end{array}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$
- Solves this subproof; return one layer

---

$$\dots \quad [f : \varphi \rightarrow \varepsilon; x : \beta] \vdash f \ x : \varphi$$

---

$$[f : \delta; x : \beta] \vdash (f \ (f \ x)) : \varepsilon$$

---

$$[x : \beta] \vdash (\text{fun } f \text{ -> } f \ (f \ x)) : \gamma$$

---

$$[ ] \vdash (\text{fun } x \text{ -> fun } f \text{ -> } f \ (f \ x)) : \alpha$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:  $\{\varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$
- Need to satisfy constraint  $\gamma \equiv (\delta \rightarrow \varepsilon)$
- Solution:  $\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta)$

---

$$\frac{\frac{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}{[x : \beta] \vdash (\text{fun } f \text{ -> } f (f x)) : \gamma}}{[ ] \vdash (\text{fun } x \text{ -> fun } f \text{ -> } f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:

$$\{\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$$

- Solves subproof; return one layer

---

$$\frac{\frac{[f : \delta; x : \beta] \vdash (f (f x)) : \varepsilon}{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}}{[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma); \gamma \equiv (\delta \rightarrow \varepsilon)$

# Type Inference - Example

- Current subst:

$$\{\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$$

- Need to satisfy constraint  $\alpha \equiv (\beta \rightarrow \gamma)$

- Solution:  $\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta))$

$$\frac{[x : \beta] \vdash (\text{fun } f \text{ -> } f (f x)) : \gamma}{[\ ] \vdash (\text{fun } x \text{ -> fun } f \text{ -> } f (f x)) : \alpha}$$

- $\alpha \equiv (\beta \rightarrow \gamma)$

# Type Inference - Example

- Current subst:

$$\{\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta)),$$

$$\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$$

- Solves subproof; return one layer

$$\frac{[x : \beta] \vdash (\text{fun } f \rightarrow f (f x)) : \gamma}{[\ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha}$$

# Type Inference - Example

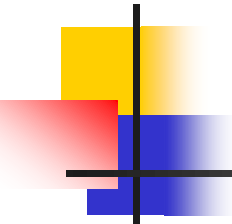
- Current subst:

$$\{\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta)),$$

$$\gamma \equiv ((\beta \rightarrow \beta) \rightarrow \beta), \varepsilon \equiv \beta, \zeta \equiv \beta, \varphi \equiv \beta, \delta \equiv \beta \rightarrow \beta\}$$

- Done:  $\alpha \equiv (\beta \rightarrow ((\beta \rightarrow \beta) \rightarrow \beta))$

$$[ ] \vdash (\text{fun } x \rightarrow \text{fun } f \rightarrow f (f x)) : \alpha$$



# Type Inference Algorithm

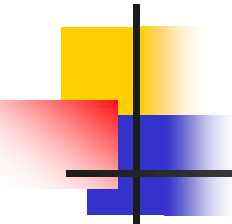
---

Let  $\text{infer}(\Gamma, e, \tau) = \sigma$

- $\Gamma$  is a typing environment (giving polymorphic types to expression variables)
- $e$  is an expression
- $\tau$  is a monomorphic type (with type variables)
- $\sigma$  is a substitution of types for type variables
- Idea:  $\sigma$  is the constraints on type variables necessary for  $\Gamma \vdash e : \tau$
- Should have  $\sigma(\Gamma) \vdash e : \sigma(\tau)$

# Type Inference Algorithm

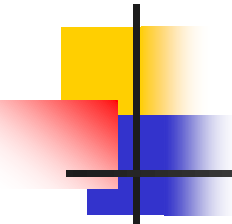
- $\text{infer}(\Gamma, \text{exp}, \tau) =$
- match  $\text{exp}$  with
  - $\text{Var } v \rightarrow \text{return Unify}\{\tau \equiv \text{freshInstance}(\Gamma(v))\}$ 
    - Replace all quantified type vars by fresh ones
  - $\text{Const } c \rightarrow \text{return Unify}\{\tau \equiv \text{freshInstance } \varphi\}$   
where  $\Gamma \vdash c : \varphi$  by the constant rules
  - $\text{fun } x \rightarrow e \rightarrow$ 
    - Let  $\alpha, \beta$  be fresh variables
    - Let  $\sigma = \text{infer}([x : \alpha] + \Gamma, e, \beta)$
    - Return  $\text{Unify}(\{\sigma(\tau) \equiv \sigma(\alpha \rightarrow \beta)\}) \circ \sigma$



# Type Inference Algorithm (cont)

---

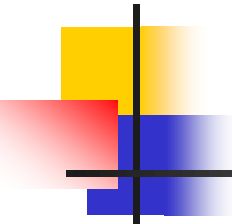
- match  $exp$  with ...
  - App ( $e_1$   $e_2$ )  $-->$ 
    - Let  $\alpha$  be a fresh variable
    - Let  $\sigma_1 = \text{infer}(\Gamma, e_1, \alpha \rightarrow \tau)$
    - Let  $\sigma_2 = \text{infer}(\sigma_1(\Gamma), e_2, \sigma_1(\alpha))$
    - Return  $\sigma_2 \circ \sigma_1$



# Type Inference Algorithm (cont)

---

- match  $exp$  with ...
  - if  $e_1$  then  $e_2$  else  $e_3 \rightarrow$ 
    - Let  $\sigma_1 = \text{infer}(\Gamma, e_1, \text{bool})$
    - Let  $\sigma_2 = \text{infer}(\sigma_1(\Gamma), e_2, \sigma_1(\tau))$
    - Let  $\sigma_3 = \text{infer}(\sigma_2 \circ \sigma_1(\Gamma), e_3, \sigma_2 \circ \sigma_1(\tau))$
    - Return  $\sigma_3 \circ \sigma_2 \circ \sigma_1$



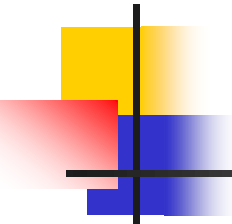
# Type Inference Algorithm (cont)

---

- match  $exp$  with ...
  - let  $x = e_1$  in  $e_2 \rightarrow$ 
    - Let  $\alpha$  be a fresh variable
    - Let  $\sigma_1 = \text{infer}(\Gamma, e_1, \alpha)$
    - Let  $\sigma_2 =$   
 $\text{infer}([x:\text{GEN}(\sigma_1(\Gamma), \sigma_1(\alpha))] + \sigma_1(\Gamma),$   
 $e_2, \sigma_1(\tau))$
    - Return  $\sigma_2 \circ \sigma_1$

# Type Inference Algorithm (cont)

- match  $exp$  with ...
  - let rec  $x = e_1$  in  $e_2 \rightarrow$ 
    - Let  $\alpha$  be a fresh variable
    - Let  $\sigma_1 = \text{infer}([x : \alpha] + \Gamma, e_1, \alpha)$
    - Let  $\sigma_2 =$   
 $\text{Infer}([x : \text{GEN}(\sigma_1(\Gamma), \sigma_1(\alpha))] + \sigma_1(\Gamma),$   
 $e_2, \sigma_1(\tau))$
    - Return  $\sigma_2 \circ \sigma_1$



# Type Inference Algorithm (cont)

---

- To infer a type, use `type_of`
- Let  $\alpha$  be a fresh variable
- `type_of` ( $\Gamma, e$ ) =
  - Let  $\sigma = \text{infer}(\Gamma, e, \alpha)$
  - Return  $\sigma(\alpha)$
- Next time: an algorithm for unification